UNDERSTANDING PERCEIVED QUALITY THROUGH VISUAL REPRESENTATIONS

A Thesis
Presented to
The Academic Faculty

by

Dogancan Temel

In Partial Fulfillment
of the Requirements for the Degree
Doctor of Philosophy in the
School of Electrical and Computer Engineering

Georgia Institute of Technology
December 2016

UNDERSTANDING PERCEIVED QUALITY THROUGH VISUAL REPRESENTATIONS

Approved by:

Professor Ghassan AlRegib, Advisor
School of Electrical and Computer Engineering
Georgia Institute of Technology

Professor Anthony J. Yezzi
School of Electrical and Computer Engineering
Georgia Institute of Technology

Professor James H. McClellan
School of Electrical and Computer Engineering
Georgia Institute of Technology

Professor Nagi Gebraeel
H. Milton Stewart School of Industrial and Systems Engineering
Georgia Institute of Technology

Professor David V. Anderson
School of Electrical and Computer Engineering
Georgia Institute of Technology

Date Approved: October 18, 2016

I dedicate this thesis to,

***My Mother Esin**,*

An idealistic soul who taught me to empathize and care,

***My Father Adil**,*

A selfless role model who taught me to help others,

***My Sister Zeynep**,*

A source of happiness who always reminds me to smile.

ACKNOWLEDGEMENTS

I am grateful to work with Prof. Ghassan AlRegib, who is not just a PhD advisor and a mentor to me, but also a friend and a family member. Our technical discussions guided me through my research and philosophical discussions guided me through my life. Thanks to the supervision of Prof. AlRegib, Multimedia and Sensors Laboratory was like a second home to me. I want to thank all former and current members of the Multimedia and Sensors Laboratory for being my colleagues, friends, and family, without whom, a research lab would be just bricks and walls. I am especially thankful to our current members Yazeed Alaudah, Motaz Al-Farraj, Chih-Yao Mao, Zhen Wang, Min-Hun Cheng, Yuting Hu, Amir Shafiq, Tariq Alshawi, Mohit Prabhushankar, and our previous members Masshour Solh, Mingyu Chen, and Michael Santoro. A special thanks goes to Mohammed Aabed, without whom this PhD journey would be incomplete. I also want to thank Dr.Arthur Redfern and Tarek Lahlou for their guidance and friendship.

I would like to thank Prof. McClellan, Prof. Anderson, Prof. Yezzi, and Prof. Gebraeel for serving on my dissertation committee. I also want to extend my thanks to all my friends and family who were always there for me. I would like to express my gratitude to my brother Yusuf Bugra Erol, my brother-in-law Andac Lulec, my source of joy Ayse Selin Cakmak, my cousins Berkay Savci, Rana Kalkan, Kubra Kalkan Cakmakci, and Tevfik Gocmen, my confidant Gamze Koseoglu, Meltem Sezer, and Gokcin Cinar, my study buddies, neighbors, and friends Didem Pehlivanoglu and Ezgi Karabulut, my good friends Caner Bayram, Can Gokler, Beril Besbinar, Deniz Uysal, Efe Yarbasi, Akansel Cosgun, Can Erdogan, Emre Yilmaz, Alper Yildirim, Bige Unluturk, Ozan Bicen, Melih Turkseven, and Kaya Demir. A special thanks goes

to my friend Tufan Akba, without whom my first conference would be a frustration rather than an amazing experience.

Before joining Georgia Tech, I did not know anyone in Atlanta. However, thanks to the Turkish Student Organization (TSO) at Georgia Tech, I had another family just after landing at the airport. TSO members helped me with everything from accommodation and transportation to course selection. The help provided by TSO was `priceless`. I became a member of TSO family and contributed as much as I can but it is never enough. I would like to thank everyone who contributes to organizations like TSO. What you consider as a little help can be a game changer for someone else. Please keep helping others, serving community, and making the world a better place to live in.

PhD is a long journey and the people around us have a key role in this journey. There is a plethora of struggles during PhD and research is just one of the struggles. During this journey, I was hospitalized, had been to an emergency room, and heard a doctor telling me I would die if I don't have a surgery. However, everything turned out to be fine. I cannot thank enough to my sister who helped me recover and get back to my PhD journey as soon as I can, also to my parents and friends. Without these amazing people around me, PhD would just be a degree rather than a journey.

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

SUMMARY

The formatting of images can be considered as an optimization problem in which the cost function is an image quality assessment algorithm. There is a trade-off between bit budget per pixel and perceived quality. Therefore, we can maximize the perceived quality and minimize the bit budget if and only if we can understand and measure the perceived quality.

In this thesis, we focus on understanding perceived quality through visual representations, which are obtained via handcrafted and data-driven approaches. We design perceived quality estimators based on visual system characteristics, and color perception mechanisms. Specifically, we use the contrast sensitivity mechanisms in retinal ganglion cells and the suppression mechanisms in cortical neurons to partially formulate a visual system. We utilize color difference equations and color descriptors to mimic pixel-wise color perception and a bio-inspired model to formulate center surround effects. Based on these visual representations, we introduce two novel image quality estimators `PerSIM` [1] and `CSV` [2], and a new image quality-assistance method `BLeSS` [3].

We combine our findings from visual system characteristics and color perception with data-driven approaches to directly obtain visual representations and measure their contribution to perceived quality. The majority of existing data-driven methods require subjective scores or degraded images in the training. In contrast, we follow an unsupervised approach trained with generic images. We introduce a novel unsupervised image quality estimator `UNIQUE` [4]. Moreover, we extend `UNIQUE` with multiple models and layers to obtain `MS-UNIQUE` [5] and `DMS-UNIQUE`. In addition to proposing image quality estimators, we analyze the role of spatial pooling strategy selection

through a comprehensive study [6]. Moreover, we analyze the effect of boosting in image quality assessment [7]. Existing studies propose a single boosted quality estimator. On the contrary, we investigate the generalizability of multi-method fusion as a framework. In addition to the support vector machines that are commonly used in the multi-method fusion studies, we propose using neural networks for boosting.

CHAPTER I

INTRODUCTION

In recent years, images have dominated online media and social networks. Users capture more images than they can process instantly because of the advances in capturing, storage, streaming, and display technologies and as a consequence, every day, billions of photos are uploaded to online platforms including Facebook ®, Whatsapp ®, Instagram ®, and Snapchat ® [8]. In general, the resolution of the images in these applications is low compared to the resolution supported by the display systems. However, as hardware increasingly supports higher resolutions (recently up to 4K ultra high definition [9]), image sharing applications are also expected to provide higher resolutions. Meanwhile, users do not want to pay more for the data, which restricts the network bandwidth usage [10]. Therefore, the challenge is increasing image quality while maintaining the bandwidth requirements, which can be considered as an image optimization problem. In this optimization problem, quality of experience (QoE) by the end user can be considered as a cost function. The definition of QoE depends on the application and in the case of imaging applications, the core of QoE is image quality. The ever-increasing number of images makes it impossible to assess the perceived quality of all publicly available images subjectively. Therefore, there is an emerging need to automatically assess the perceived quality of images.

The ideal way to estimate the quality of images is by subjective evaluation. However, subjective evaluation is too demanding in terms of resources and time. Therefore, objective methods are designed to estimate image quality. The main challenge in image quality assessment is the problem definition because it is not intuitive to define quality. If there is a distortion-free reference image, quality of a distorted image can

be measured by quantifying pixel-wise differences between the distorted image and the distortion-free reference image. This type of quality definition is denoted as fidelity. Fidelity-based approaches have dominated image quality assessment research for a long time. However, in recent years, the research community has started to pay more attention to perception and its role in defining quality. The objective of perceptual quality assessment is to focus on subjects' perception and their quality of experience rather than pixel-wise fidelity.

Perception of images is not solely affected by the constraints caused by physical systems such as acquisition and display, but also by the degradations caused by storage (compression) and transmission (streaming). In addition to the processes from acquisition to display, the characteristics of the human visual system need to be understood to comprehensively model the perceived quality assessment. All these factors contributing to the final perception of a visual stimuli can be combined under a single pipeline we denote as `Pixels to Perception (P2P)`, which is shown in Fig. 1. The `P2P` pipeline starts with capturing an image. This image is then stored in digital platforms and transferred to other devices. Eventually, the image is perceived by end users when displayed, which ends the `P2P` pipeline.

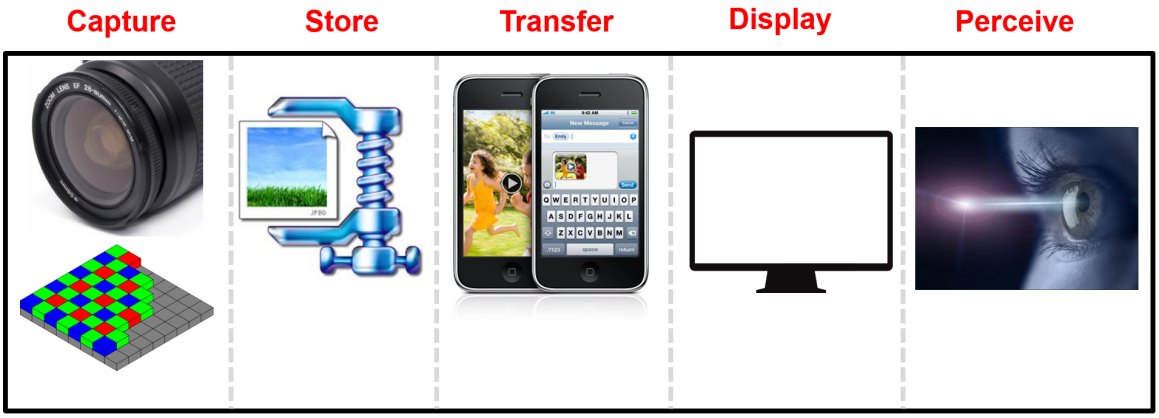


Figure 1: Pixels to perception (P2P) pipeline in practice.

We can model acquisition, storage, communication, and display processes in the P2P pipeline given the conditions and the configurations of hardware and software setup. However, perception processes in a visual system are still not fully understood. It is not sufficient to study the sensory experience to model the perceptual experience of the subjects because visual information is processed after the acquisition stage. Therefore, we need to understand the basics behind the perception process to comprehensively measure the perceived quality. In contrast to neuroscientists, psychophysicists, and other experts, we approach the perceived quality assessment problem from an engineering point of view and focus on mapping pixels to perception. We can consider this mapping as a transfer function and learn the characteristics of this function by feeding inputs and observing outputs of a system. The inputs are images and the outputs are scores assigned by the subjects as illustrated in Fig. 2.

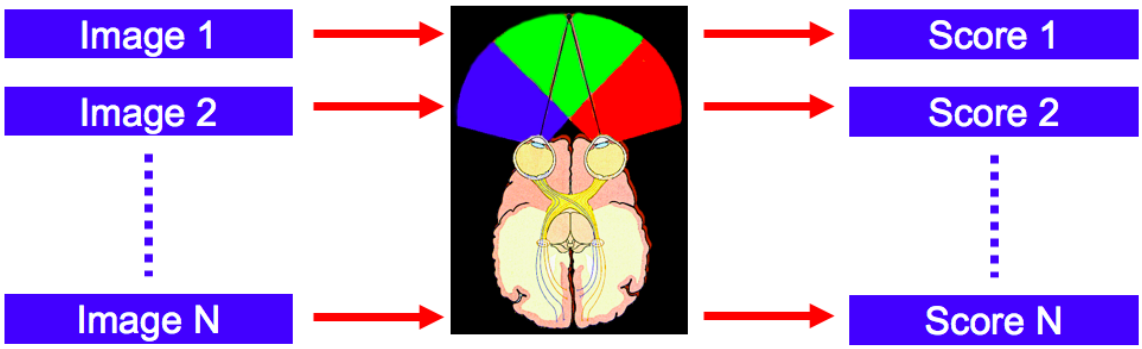


Figure 2: Mapping images to scores in the (P2P) pipeline.

The pipeline of a full-reference image quality assessment algorithm that maps pixels to perception is summarized as in Fig. 3. In general, there are three main blocks in the mapping process. The first block is responsible for visual representation generation. The type of the visual representation depends on the algorithm. The second block is responsible for comparison of the visual representations and the third block is responsible for pooling compared representations to obtain a quality score. In this thesis, we start with describing visual representations based on visual system

characteristics and color perception. We utilize similarity or dissimilarity measures to compare these visual representations, and combine various representations to obtain quality estimators. Furthermore, we combine our findings from visual system characteristics and color perception with unsupervised learning techniques to propose a visual representation generation mechanism. To investigate the role of the pooling block in an image quality assessment algorithm, we perform a comprehensive study of spatial pooling strategy selection. The aforementioned research directions cover all three main blocks in the image quality assessment algorithm pipeline. However, the performance of a single algorithm may not be sufficient. To obtain strong image quality assessment algorithms from weak ones, we also analyze the effect of boosting in image quality assessment.

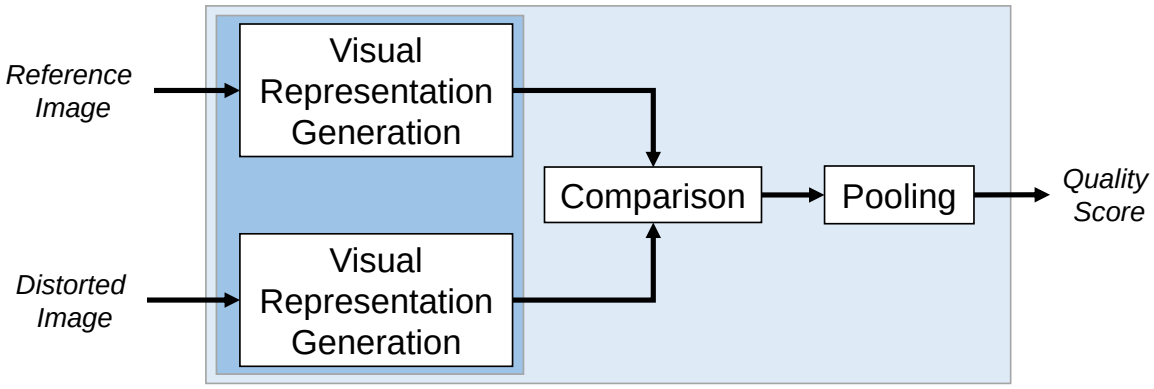


Figure 3: Image quality assessment algorithm pipeline.

CHAPTER II

LITERATURE REVIEW

In this chapter, we introduce the literature of designing image quality assessment algorithms and describe the relation of this thesis to the literature. We classify the literature into four subjects: handcrafted image quality estimators, data-driven image quality estimators, spatial pooling strategy selection, and boosting-based image quality estimators.

2.1 Handcrafted Image Quality Estimators

We separate handcrafted image quality estimators into two classes. The first class includes estimators based on fidelity, structure, and scale space. The second class contains methods based on color, visual system, and spatial pooling.

2.1.1 Fidelity, Structure, and Scale Space

Fidelity attributes quantify the changes in a degraded image with respect to a reference image and these attributes are commonly preferred in image and video coding standards for rate-distortion optimization because of their low computational complexity and ease of implementation. An intuitive method to measure the fidelity of an image is to directly compare it with a distortion-free image, if available. Mean squared error (MSE) is a commonly used pixel-wise fidelity method, which is calculated by obtaining the difference between images, taking the square root of the difference, and calculating the mean value. MSE is scaled by the range of an image and mapped using a logarithmic function to obtain the peak signal-to-noise ratio (PSNR). The intensity channel is commonly used to obtain PSNR but it can also be calculated

over each color channel individually and the average is obtained. In FSIMc [11], images are transformed from RGB domain to La*b* domain and pixel-wise similarity is calculated over a* and b* channels to quantify the fidelity. These pixel-wise fidelity measurements do not correlate well with the perceived quality of a visual stimulus as shown in [12, 13].

Instead of focusing on individual pixels, the human visual system perceives groups of pixels that are similar. In the image quality assessment literature, structural similarity is commonly obtained by quantifying the similarity between the mean subtracted and divisive normalized images. The authors in [14] propose a full reference method (SSIM) based on the comparison between reference and distorted images in terms of luminance, contrast, and structure in the spatial domain. These structure-based metrics are also extended to multi-scale (MS-SSIM) [14], complex domain (CW-SSIM) [15], and information-weighted (IW-SSIM) [16] versions. Fidelity-based approach PSNR is also extended with structural components leading to PSNR-HVS [17], PSNR-HVS-M [18] , PSNR-HA [19], and PSNR-HMA [19]. Mean luminance value is used in C4 [20] whereas mean subtraction and divisive normalization are used in BRISQUE [21] to obtain more descriptive feature maps compared to initial pixel maps. Locally normalized images are used in [22] to train a convolutinal neural network for no-reference image quality assessment. Instead of basic normalization operations, divisive normalization transform (DNT) is used by Li and Wang [23] after linear image decomposition and by [24] (REDLOG) over locally weighted gradient magnitudes. Scale-space representations also include inherent normalization steps but they are different from direct mean subtraction and divisive normalization.

Instead of maintaining a spatial representation, images can be transformed into different domains including but not limited to Fourier, DCT, wavelet or curvelet. The main motivation to use domains beyond the spatial domain is the human visual system, which is sensitive to the frequency of patterns in visual stimuli [25]. Moreover,

multi-scale representations are also commonly used in the computer vision and image processing community such as pyramid, which basically applies smoothing and subsampling in a recurrent fashion. As summarized in Table 1, the majority of existing image quality estimators include scale-space representations other than the methods based on residual error (MSE, PSNR, PSNRc), single level structural similarity metric (SSIM [14]), and color difference metric (CIEDE2000 [26]).

2.1.2 Color, Visual System, and Pooling

The human visual system (HVS) is more sensitive to changes in intensity compared to color as exploited in the chroma subsampling for image coding [27]. Therefore, luma channels can be more informative compared to chroma channels in terms of perceived quality. Although color may not be as informative as intensity, there is still additional information in color, that is not conveyed by intensity. An intuitive way to introduce color information into quality assessment is pixel-wise fidelity. PSNRc and FSIMc [11] introduce color information by computing pixel-wise fidelity over each channel in the RGB color space and chroma channels in the La*b* color space. However, pixel-wise fidelity approaches overlook the characteristics of color, which implies that color is not a metric space and when it is treated as such, it would lead to problems [28]. The difference between individual color channels would not necessarily correspond to the perceived difference between colors. Therefore, instead of treating color channels as equivalent and separate, we should focus on the overall perceived color as a combination of these channels.

The color science community develops formulations to quantify the color differences. The International Commission on Illumination (CIE) determines the lighting-related standards including color differences [29]. CIEDE2000 is a commonly used color difference equation introduced by CIE [30, 31]. In terms of the application field of the color difference equations, the approach in [32] is a transition from basic tone

matching to textured image comparisons. The authors in [33] discuss the connections between image quality, appearance, and color difference. In [26], the authors combine CIEDE2000 color difference with the printing industry standards for visual verification to assess the perceived image quality. Fidelity and color difference-based approaches are usually computed over the entire image. However, in the reduced-reference quality estimator C4 [20], color-based features are extracted around the characteristic points. RGB images are transformed into Krauskopf's color space and the mean value of chrominance channels are computed around these characteristic points.

Fidelity-based methods can be combined with visual system characteristics to obtain perceptually-extended image quality estimators. The authors in [17] extend PSNR by removing the mean shift, stretching the contrast block-wise, and quantizing the DCT coefficients with the compression table proposed by JPEG. These extensions are performed to make PSNR compatible with the human visual system and the extended metric is named as PSNR-HVS [17]. Reduction by value of contrast masking is also added to the metric and the modified version is named as PSNR-HVS-M [18]. These metrics are further extended by adding contrast change and mean shifting sensitivity (PSNR-HA, PSNR-HMA) as explained in [19]. The authors in [34] include wavelet-based models of visual masking and visual summation to weight the SNR map (VSNR) [34]. In SR-SIM [35], suppression mechanisms are modeled by spectral residual, which is calculated in the frequency domain. A degradation model denoted as NQM is proposed by the authors in [36] based on linear frequency distortion and additive noise injection. Frequency-based distortion measures and additive noise-based quality measures mimic the HVS by considering contrast sensitivity, local luminance, contrast interaction between spatial frequencies, and contrast masking effects.

The functional role of neurons and neural systems is investigated by the authors in [37]. More specifically, they try to develop a model for early sensory processing,

which includes the non-linearities and the adaptation mechanisms in cortical neurons. The statistics of natural scenes can be analyzed by decomposing images using basis functions. Intuitively, natural images can not be decomposed into independent components using linear basis functions because the origin of these images are not based on fusing independent patterns. Even if individual patterns could be represented as combinations of linear basis functions, linearities turn into non-linearities in case of an occlusion. Therefore, linear decomposition-based representations can only approximate natural scenes. The statistical properties of natural scenes can be extracted using the steerable pyramid [38], whose basis functions are translations, rotations, and dilations of a common filter kernel. When natural images are projected onto these basis functions, the joint statistics of these coefficients contain non-linear dependencies. It has been shown that these non-linear dependencies can be reduced using normalization operations [39, 40, 41]. These normalization operations correspond to the suppression mechanisms in a visual system [40]. The divisive normalization transform is used in image quality assessment methods to reduce spatial redundancies in visual representations [23, 24]. Alternatively, normalization can directly be performed in the spatial domain. In learning-based computer vision applications, images are fed to normalization blocks to filter out redundancies and keep distinctive features. A commonly used architecture in these learning methods is the convolutional neural network (CNN), which contains normalization layers. When CNNs are used for object recognition, a global normalization is applied over an entire image to avoid saturation, illumination, and contrast variation issues. In the case of image quality assessment, local normalization outperforms global normalization [22].

In the image processing and the computer vision literature, difference of Gaussian or Laplacian of Gaussian operators are commonly used to obtain local descriptors. These descriptors extract the bandpass information that can characterize images in a more distinctive manner compared to original pixel values. Therefore, these operators

are commonly used in applications including but not limited to classification and image retrieval. The authors in [42] show that the contrast sensitivity of retinal ganglion cells of a cat can be modeled with a difference of Gaussian formulation. Similarly, a Gaussian derivative-like approach is proposed by the authors in [43] to model neural mechanisms in the human foveal retinal vision and it is claimed to outperform Gabor filters based on model-free Wiener filter analysis.

Multi-scale representations and transforms described in Section 2.1.1 can be considered as partial visual system models because neural responses in a visual cortex include scale-space orientation decomposition. Computational models can also be used to mimic the behavior of the visual system. The authors in [44] combine source, distortion, and HVS models to obtain the image information measure VIF. Source images are modeled with Gaussian scale mixtures and distortion model is based on signal attenuation and additive noise in wavelet domain. The HVS is modeled as a distortion channel with stationary, zero mean, and additive white Gaussian noise in the wavelet domain. A feature similarity index (FSIM) is proposed by the authors in [11], which partially models low-level feature perception in a visual system using phase congruency (PC) and gradient magnitude (GM). PC consists of a log-Gabor filter and a Gaussian spread function, and GM is based on gradient operators.

Quality maps or descriptors are usually pooled with sum, difference, divergence, min, max, mean, and weighted sum without further discussing other pooling strategies. No-reference method BRISQUE [21] uses regression-based mapping function learned from data. CW-SSIM [45] and IW-SSSIM [16] are the only approaches that comprehensively investigate and explicitly discuss alternative pooling strategies.

Table 1: A comparative analysis of objective image quality assessment methods.

Name	Year	Type	Fid.	Mean subt.	Div. norm.	Col.	Sc. sp.	Visual system	Pooling
MSE,PSNR		FR	+						Mean over feature map
PSNRc		FR	+			+			Mean over feature map
NQM [46]	2000	FR					+	Contrast sesitivity Contrast masking	Sum over feature map
MS-SSIM [14]	2003	FR		+	+		+	Contrast masking Weber's law	Mean over feature map
SSIM [14]	2004	FR		+	+			Contrast masking Weber's law	Mean over feature map
PSNR-HVS [17]	2006	FR		+	+		+	Contrast sensitivity	Mean over feature map
PSNR-HVSM [18]	2007	FR		+	+		+	Contrast sensitivity Contrast masking	Mean over feature map
VSNR [34]	2007	FR					+	Near-threshold Suprathreshold	Sum over feature vector
VIF [44]	2008	FR					+	HVS as distortion channel	Sum over channels
C4 [20]	2008	RR		+		+	+	Eyes, V1, V2, Ventral pathway, Cortex filters	Mean over feature map
CW-SSIM [45]	2009	FR					+	Primary visual cortex	Mean, median min, max over feature map
Li and Wang [23]	2009	RR			+		+	Visual masking and visual cortex cells	KL divergence and difference between feature vectors
PSNR-HA [19]	2011	FR		+	+		+	Contrast sensitivity	Mean over feature map
PSNR-HMA [19]	2011	FR		+	+		+	Contrast sensitivity Contrast masking	Mean over feature map
FSIM [11]	2011	FR					+	Human visual cortex	Weighted sum over feature vector
FSIMc [11]	2011	FR	+			+	+	Human visual cortex	Weighted sum over feature vector
IW-SSIM [16]	2011	FR		+	+		+	Contrast masking Weber's law Local sensitivity	Information weighted sum over feature map
CIEDE2000 [26]	2012	FR				+		Color perception	Mean over difference map
BRISQUE [21]	2012	NR		+	+		+	Contrast gain masking	Regression-based mapping
SR-SIM [35]	2012	FR					+	Suppression mechanisms	Weighted sum over feature maps
REDLOG [24]	2015	RR			+		+	Contrast sensitivity visual cortex cell	Weighted differ. between features

2.1.3 Summary

The analysis of quality attributes and pooling strategies used in various state of the art approaches is summarized in Table 1, which includes the name of the method, the year it was introduced, visual system-based characteristics, and spatial pooling strategies. Moreover, we classify the methods based on pixel-wise fidelity (fid.), mean subtraction (mean subt.), divisive normalization (div. norm.), color (col.), and scale-space representation (sc. sp.). There is a (+) mark in the table cell if a specific method contains the corresponding attribute otherwise there is no mark. We also include the type of the quality estimators, which can be grouped into three different classes according to the input characteristics as follow:

- Full-reference (FR): requires an existing reference image and a distorted image,
- Reduced-reference (RR): requires certain attributes from a reference and a distorted image, and
- No-reference (NR): solely relies on a distorted image and these methods are usually based on prior knowledge of the image and the distortion.

Fidelity-based approaches can accurately detect the numeric changes in images but these fidelity-based measurements are not highly correlated with perceived quality. Scale-space representations including but not limited to wavelets and curvelets are already well-studied in image quality assessment literature. Pixel-wise chroma fidelity, color difference formulations, and color-based feature extraction around characteristic points are used to estimate color degradations. The approaches based on fidelity and feature extraction overlook the characteristics of color. Even though color difference equations consider the perceptibility of colors, they measure the degradation pixel by pixel and ignore the center-surround effects.

Contrast-related characteristics including sensitivity and masking are commonly used by image quality estimators. Weber's law is used by the structural methods

to measure relative changes instead of absolute changes. Some of the image quality estimators claim to partially model visual system without specifying the exact functionality or the accuracy of the model. The pooling strategy selection is usually overlooked compared to the efforts in the quality attribute design.

2.2 Data-driven Image Quality Estimators

We separate data-driven image quality estimators into four classes. The fist class includes estimators based on natural scene statistics, the second class contains methods based on dictionary and filter learning, and the third class includes neural network-based methods. In the forth class, we include data-driven methods that do not belong to the first three classes.

2.2.1 Natural Scene Statistics-based Methods

Statistical characteristics of images can be utilized to directly obtain quality estimators. The authors in [47] measure distortions through statistical features that are based on natural image statistics, distortion texture statistics, and blur/noise statistics. A principal component analysis is used to decrease the dimension of the features and a support vector machine is used for each feature type to obtain a feature-based regression formulation. Regressed outputs are linearly combined to obtain a score (L-BIQ). In [48], the authors feed LBIQ features to a deep belief network with a ReLU non-linearity and the output of the network is regressed with a Gaussian process to obtain a quality estimate. The authors in [21] propose a no-reference image quality assessment method BRISQUE based on natural image statistics in the spatial domain. Luminance channels of the images are locally normalized by mean subtraction and divisive normalization. After normalization, intensity values are multiplied with the neighboring pixels in horizontal, vertical, and diagonal directions to obtain the distribution of these terms. Experimental distributions are matched with asymmetrical generalized Gaussian distributions to obtain the distribution related parameters.

Regression is used to map the features to quality scores. In [49], the authors propose a two-stage framework denoted as DIIVINE, which corresponds to a distortion identification stage and a distortion-specific quality assessment stage. The statistical features are based on variance of subband coefficients, shape parameter of subband coefficients, shape parameter across subband coefficients, correlations across scales, spatial correlation across subbands, and across orienation statistics. These statistical features are obtained through fitting generalized Gaussians, fitting polynomials or calculating correlations. A support vector machine is used in the distortion identification stage and a support vector regression is used in the distortion-specific quality assessment stage.

2.2.2 Dictionary- and Filter Learning-based Methods

In addition to natural scene statistics-based methods, dictionary and filter learning methods are also proposed in the literature. The authors in [50] propose a quality estimator denoted as CORNIA, which follows an unsupervised learning approach in the dictionary learning stage and a supervised approach in the regression stage. In the unsupervised learning approach, a dictionary is learned from distorted images, which consist of synthesized distortions based on speckle noise, Poisson noise, salt-pepper noise, and zero-mean Gaussian white noise. Moreover, distorted images from the CSIQ database including JPEG compression, JP2K compression, additive pink Gaussian noise, and Gaussian blur are also utilized in the dictionary learning. K-means clustering is used in the dictionary generation, soft assignment coding in the local feature encoding, max-pooling in the fixed-length feature vector generation, and regression in learning the mapping between features and scores. The authors in [51] propose a no-reference image quality assessment approach based on filter learning, which is achieved by a joint optimization stage. The set of filters and the prediction model are obtained simultaneously through a supervised learning approach. The

weights are learned with a support support vector regression mechanism and the filter set is learned with a stochastic gradient descent approach. As an alternative to the supervised learning-based filter set, k-means clustering-based filters are also used. Global features are computed from the maximal and the minimal values of filter responses and regression is used to obtain a quality score. The authors in [52] propose an image quality assessment method based on quality-aware filters (QAF). Mean subtracted and contrast normalized coefficients and responses of Gabor filters are used to obtain local descriptors. A sparse filtering operation is performed over these descriptors to obtain quality aware filters. These filters are used to encode local descriptors and max pooling is performed to obtain quality aware features. A random forest is trained to regress these features to quality scores. The authors in [53] propose a sparse representation-based image quality assessment method denoted as SPARQ. A saliency detection algorithm is used to detect visually important patches, sparse coefficients are computed by decomposing these patches with a dictionary, and these sparse coefficients are compared to measure the quality of a distorted image. Reference images are used to obtain an overcomplete dictionary, which is achieved with a sparsity constraint.

2.2.3 Neural Network-based Methods

The authors in [22] propose a no-reference image quality assessment method based on convolutional neural networks (CNN). In the training phase, sample patches are extracted from training images and these patches are feedforwarded in the network to obtain patch descriptors. Error signals are calculated using the high level descriptors and the labels. Then, weights in the fully connected layers and the kernels in the convolutional layers are adjusted using backpropagation. Grayscale images are used for training and testing, and color information is not utilized. In [54], in addition to estimating the perceived quality, CNNs are also used to identify the distortion type.

The previous CNN architecture [22] is extended by adding a multi-class logistic-regression layer for classification, increasing the number of convolutional layers, and reducing the receptive field of the filters and the number of neurons in the fully connected layers. In [55], the authors propose obtaining image quality scores through a weighted sum of patch-based CNN scores. At first, an image is segmented with a graph-based approach and the gradient map of the segmented image is computed with the Prewitt operator. For each patch in a processed image, the corresponding entities in the gradient map are summed to weight the CNN score. Therefore, instead of directly averaging the CNN scores, a weighted average based on the gradient of the segmented image is used. The authors in [56] use CNNs with different design choices ranging from pre-trained to fine-tuned networks in image quality assessment. CNNs are used for feature extraction and support vector machine-based regression is used for quality score generation. Pre-trained CNNs are obtained from image classification databases including ImageNet [57], Places [58], and a hybrid of both. The fully connected layers of these networks are substituted with randomly initialized layers and trained with images and their subjective scores.

2.2.4 Other methods

In [59], the authors propose a learning-based full reference image quality estimator MLIQM, whose features are based on luminance, contrast, structure, chrominance, colorimetric dispersion, and steerable pyramids. A support vector machine is used for quality level classification and a final score is obtained by support vector regression. In [60], the authors propose two image quality assessment methods based on the area (Q_{area}) and the curvature ($Q_{exponent}$) of image reciprocal singular value curves. These image quality assessment methods can classify the processed images based on the noise variance as noise or non-noise, because the characteristics of the curves are different for noise class and non-noise class. Distortion identification stage is used to

set the threshold parameters in the quality estimators. In [61], the authors propose a learning-based quality assessment approach, which does not require subjective scores. Instead of subjective scores, percentile-pooled FSIM scores [11] are used. In the training stage, reference and distorted images are divided into overlapped patches and the similarity between patches are computed through FSIM. Local FSIM scores are normalized with global percentile-pooled FSIM. Patches are grouped based on their FSIM scores and a high pass filtering operation is performed over the patches to obtain local structures. Then, these patches are clustered with respect to their local structures, which is denoted as quality aware clustering (QAC). A weighted average-based formulation of the distance with respect to the centroid of these clusters is used to estimate the quality. To obtain quality-aware clusters, images in the training set are randomly selected from the Berkeley [62] database and these images are degraded with distortion types including Gaussian noise, Gaussian blur, JPEG compression, and JPEG2000 compression. The authors in [63] follow a fuzzy inference-based approach to formulate subjective quality assessment. First, natural scene statistics-based features are extracted. Second, subjective quality assessment is modeled as a fuzzy process by finding the relationship between subjective scores and discrete quality levels (bad, poor, fair, good, and excellent). Then, the mapping between discrete quality levels and natural scene statistics-based features are learned through a semi-supervised locally linear embedding method, which uses both the labeled and the unlabeled images. In [64], the authors also map natural scene statistics-based features to discrete quality levels. To classify images into different quality levels, a deep belief net is used in the pre-training and back-propagation is used in the fine-tuning. The authors in [65] propose using preference image pairs (PIPs) in image quality assessment. A PIP refers to an image pair, in which one of the image is preferred to other one in terms of perceived quality. In [65], the scope of the study is limited

to images that are easily distinguishable in terms of perceived quality. The PIP generation procedure starts with collecting images that are diverse in terms of content, distortion type, and distortion level. Pairs are randomly selected from the collected images and subjects assign preference labels for each pair. PIPs are generated from existing image quality databases and the proposed method in [65] is tested in these PIPs. DCT statistics [21], L-moments [66], and sparse representation-based natural scene statistics [67] are fused to obtain image-based features. A difference feature is obtained by computing the difference between the features of compared images. To map the difference features to image preference labels, a supervised approach is followed based on a multiple kernel learning method [68].

2.3 Spatial Pooling Strategy Selection

There has been a significant effort in engineering image quality assessment methods. The majority of the previous studies focused on the feature design part and overlooked spatial pooling strategy selection, as briefly discussed in the last paragraph of Section 2.1.2. There are several studies in the literature that investigate the effect of spatial pooling strategy selection. The authors in [69] investigate the effect of spatial pooling strategies for pixel-wise and structural image quality methods. Mean, Minkowski, quality/distortion weighted, and information-weighted pooling strategies are used in the comparison. The types of distortions used in the experiments are compression, image noise, communication, and blur. In [70], the authors propose two spatial pooling techniques. The first pooling strategy is based on the fixation of a visual system to predict where an average observer looks. Fixation is computed using an algorithm denoted as gaze-attentive fixation finding engine (GAFFE) [71]. GAFFE selects the center of an image as the first fixation and foveates around the center using luminance, contrast, luminance-bandpass, and contrast-bandpass features to obtain a fixation map, which is used to weight quality maps. The second pooling

strategy is based on the fact that significant degradations over images dominate the perceived quality. To model this observation, highly distorted pixels are scaled before spatial pooling, which is denoted as percentile pooling. Compression, image noise, communication, and blur are used to degrade test images.

The authors in [72] evaluate existing pooling strategies for color printing, in which different rendering methods are used to print images, which are scanned to be used by objective image quality methods. The methods that are used for spatial pooling are mean pooling, Minkowski pooling, quality/distortion weighted pooling, information-weighted poling, frequency-tuned saliency-weighted pooling (IG) [73], and nonparametric bottom-up saliency model-based pooling (NB) [74, 75]. SSIM [14], S-CIELAB [76], S-DEE [77], WLF [78], ABF [79], and δLC [80] attributes are used to obtain quality/distortion maps. In [81], the authors propose a pooling strategy denoted as 5-Number summary (5-N), which is based on combining percentile values with min, median, and max pixel values. The performance of 5-N is compared with spatial pooling strategies including mean, Minkowski, quality/distortion weighted, and percentile. SSIM is used as the quality attribute and confidence intervals of correlation coefficients are provided in the results. Degradation types in the image set include compression, image noise, communication, blur, color, global, and local. The authors in [82] propose a spatial pooling strategy based on luminance-contrast (L-C) dependence. L-C pooling is compared with mean pooling, information-weighted pooling, and fixation-based pooling. SSIM is used to obtain quality maps and images are degraded with distortion types including compression, image noise, communication, blur, global, and local. In [83], the authors propose obtaining a spatial pooling strategy SP by training support vector machines with histograms and statistical descriptors including mean, standard deviation, and 5-N. SP is compared with mean pooling, standard deviation (std) pooling, and coefficient of variation (CoV) pooling.

SSIM, gradient magnitude similarity, and FSIM are used to obtain quality maps. Images are degraded with compression, image noise, communication, blur, global, and local distortion categories.

2.4 Boosting-based Image Quality Estimators

The majority of the quality estimators refer to visual system characteristics but none of them is a comprehensive model of the perception process. Existing image quality estimators differ from each other in various ways. However, all these methods fundamentally map pixels to subjective scores. Moreover, even some of the methods are less perceptually correlated than others, they can still contain additional information that can not be provided by better performing methods. Therefore, multiple methods can be fused to boost the overall performance. Boosting is initially discussed in [84] and [85] to investigate whether it is possible to obtain strong learners from weak learners or not. In [86], the authors describe a method for converting a weak learning algorithm into a strong one that obtains arbitrarily high accuracy.

Based on the boosting discussion, we can also convert image quality assessment algorithms with poor performance into highly perceptually correlated quality estimators. In [87], the authors use canonical correlation along with linear regression to obtain a quality estimator by fusing four quality estimators that are based on fidelity and structure. The authors in [88] propose a regression-based approach that is used to non-linearly fuse quality estimates of multiple methods. In addition to estimating a quality score directly, hand-crafted features are also used to classify distortion types and a regression approach is used in each distortion type separately to learn the mapping function. In [89], the multi-method fusion is extended with a method selection algorithm to reduce the overall complexity. The authors in [90] follow a regression-based approach to obtain two types of image quality estimators that are

separately trained with features of existing quality estimators and hand-crafted features that measure degradations overlooked by the existing features. The scores of individual quality estimators are fused with a support vector regression stage along with a statistical testing-based selection mechanism. Boosting is also used for video quality assessment [91] and stereo image quality assessment [92].

2.5 Summary

In this section, we summarize the literature of designing image quality assessment algorithms and describe the relation of this thesis to the literature.

Table 2: Characteristics of hand-crafted image quality estimators.

YEAR	Before 2000		2000	2003	2004	2006	2007		2008		2009		2011					2012			2014	2015
QUALITY ESTIMATORS	MSE-PSNR	PSNRc	NQM	MSSSIM	SSIM	PSNRHVS	PSNRHVSM	VSNR	VIF	C4	CWSSIM	Li-Wang	PSNRHA	PSNRHMA	FSIM	FSIMc	IWSSIM	CIEDE	BRISQUE	SR-SIM	CNN	REDLOG
Fidelity	■	■																				
Structure				■	■	■	■			■		■	■	■			■		■		■	■
Scale Space			■	■		■	■	■	■		■	■	■	■	■	■	■		■	■		■
Visual System			■	■	■	■	■	■	■	■	■	■	■	■	■	■	■	■	■	■		
Pooling											■						■					
Color		■								■						■		■				

2.5.1 Hand-crafted Image Quality Estimators

In this thesis, we classify hand-crafted image quality estimators in the literature based on the characteristics including fidelity, structure, scale-space, visual system, pooling, and color as shown in Table 2. The definitions of fidelity, scale-space, and color are the same as in Table 1. Structure refers to methods that include divisive normalization, visual system corresponds to methods that explicitly discuss the visual system functionality, and pooling refers to methods that either compare the proposed pooling strategy to others or validate the proposed pooling strategy. As discussed earlier, pixel-wise fidelity is not commonly used because it does not highly correlate with the

perceived quality. Structure and scale-space are commonly used and well-studied. Pooling strategy selection and color information are not commonly used in the literature and visual-system based studies require a better understanding of the perception process. Therefore, instead of focusing our attention on fidelity, structure, or scale-space, thesis work concentrates on visual system, color, and pooling as highlighted with light blue in Table 2.

Table 3: Characteristics of data-driven image quality estimators.

YEAR		2011		2012			2013			2014				2015					2016
EXISTING STUDIES		LBIQ	DIIVINE	CORNIA	BRISQUE	MLIQM	CB/SF	QAC	SPARQ	Tang	QAF	Kang	Q_{area} $Q_{exponent}$	IQA-CNN++	Li	S^2F^2	DLIQA	Gao	CNN-SVR
Fidelity																			
Structure			■	■	■	■	■		■		■	■		■	■				
Scale Space		■	■			■				■	■					■	■	■	
Visual system			■		■	■		■	■						■	■	■	■	
Pooling		■		■				■			■	■			■		■		■
Color						■													
Do not require	Distortion specific data in the training								■				■						
	Labels in the training							■	■				■						
	Handcrafting			■			■					■		■					■
	Reference in testing	■	■	■	■		■	■		■	■	■	■	■	■	■	■	■	■
Multiple layers/models without handcrafting												■		■	■				■

2.5.2 Data-driven Image Quality Estimators

In addition to fidelity, structure, scale space, visual system, color, and pooling, we classify data-driven methods based on their data and hand-crafting dependence as shown in Table 3. In terms of data dependence, we analyze whether a quality estimator needs distortion specific data in training, labels in training or reference images in testing or not. Moreover, we also classify based on the depth or the width of the learning architecture and the handcrafting requirement. In the analyzed data-driven methods, pixel-wise fidelity is not directly used whereas scale space, visual system, and pooling are used in some of the methods. Majority of the analyzed methods use structure and do not require a reference image. To focus on the characteristics that are not well studied, this thesis work concentrates on proposing data-driven approaches that use color and do not require handcrafting, distortion specific data or

labels in the training as highlighted with light blue in Table 3. In addition to one layer and single model learning architectures, we also study wider and deeper models that do not require handcrafting.

Table 4: Characteristics of spatial pooling strategy selection studies.

YEAR		2007	2009	2012	2014	2014	2016
EXISTING STUDIES		Wang	Moorthy	Gong	Zewdie	Bruni	Li
Spatial Pooling Strategies	Mean	■	■	■	■	■	■
	Min/Max						
	Minkowski	■		■	■		
	Quality/Distortion Weighted	■		■	■		
	Percentile Pooling		■		■		
	Information-weighted	■		■		■	
	Others		Fixation	Saliency(IG, NB)	5-N	Fixation, L-C	STD, CoV, SP
Distortion Categories	Compression	■	■		■	■	■
	Image Noise	■	■		■	■	■
	Communication	■	■		■	■	■
	Blur	■	■		■	■	■
	Color				■		
	Global				■	■	■
	Local				■	■	■
	Others			Rendering Methods			
Quality Attribute Types	Squared Error	■					
	Structural Similarity	■	■	■	■	■	■
	Others			ΔLC, ABF, S-CIELAB, S-DEE, WLF			GMS, FSIM
Statistical Significance Tests					■		

2.5.3 Spatial Pooling Strategy Selection

We classify pooling strategy selection studies in terms of used spatial pooling strategies, distortion categories, quality attribute types, and statistical significance tests as summarized in Table 4. Existing studies generally focus on pooling strategy selection issue from a limited point of view, which prevents the results from being generalizable. The majority of the existing studies do not include statistical significance tests. Therefore, we cannot conclude whether numerical changes are statistically significant or not. Moreover, using a single quality attribute type limits the reliability of conclusions because we do not know whether or not similar conclusions are valid for different quality attributes. To obtain conclusions that are generalizable, we need to perform a more comprehensive study. Thus, in this thesis, we perform a comparative study

that includes multiple quality attributes and pooling strategies along with statistical significance tests as highlighted with light blue in Table 4. Specifically, we utilize 3 quality attribute types, 8 spatial pooling strategies, and 7 degradation categories.

Table 5: Characteristics of boosting-based image quality estimators.

YEAR		2008	2011	2013	2015
EXISTING STUDIES		Liu and Yang	Liu *et al.*		
Fidelity		■	■	■	■
Structure		■	■	■	■
Scale Space		■	■	■	
Visual system		■	■	■	■
Color					■
Boosting Method	SVR		■	■	■
	Others	Canonical Correlation Analysis			

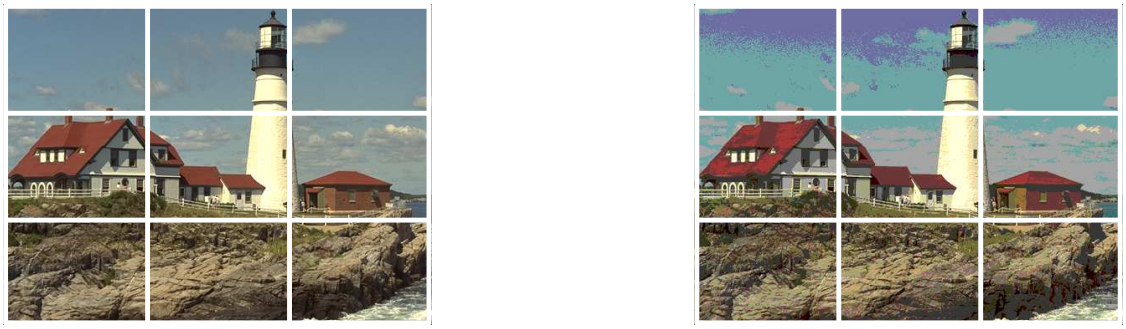
2.5.4 Boosting-based Image Quality Estimators

We classify boosting-based quality estimators in terms of the characteristics of boosted methods and boosting methods as in Table 5. Fidelity-, structure-, and visual system-based methods are used in all the analyzed boosting-based methods. Quality estimators that are based on scale space are used in all the boosting-based methods with one exception. Color-based quality estimators are only used in one of the boosting-based methods. The majority of the boosting-based methods use support vector machines for regression and only one boosting-based quality estimator use a regression technique different from support vector machines. None of the existing boosting studies reported in Table 5 use methods that contain all the characteristics. Therefore, in this thesis, we focus on boosting quality estimators that are based on all five categories, including color, and in addition to support vector regression, we focus on using an alternative technique as highlighted with light blue in Table 5.

CHAPTER III

VISUAL REPRESENTATIONS

In this chapter, we describe visual representations that can be used to understand and measure perceived quality. At first, we describe representations based on visual system characteristics. Then, we discuss representations based on color. Finally, we describe representations that are based on both the visual system and color. To exemplify the visualization of these representations, we use the TID2013 database [93] and specifically the `lighthouse2` image, which is a degraded version of the original image with quantization error. All of the images are shown with a grid structure to make visual comparisons easier. Reference and distorted images are shown in Fig. 4.



(a)Reference Image (b) Distorted Image

Figure 4: Reference and distorted versions of the lighthouse2 image from the TID 2013 database [93].

Structural degradations over the sky region are obvious in the top grids and also in the right side of the middle row. There is a significant color degradation in the upper part of the top row and also observable tone difference in sky regions. Degradations are less obvious around the highly textured regions as observed in the bottom grids, which are mostly textured rocks. In the middle row, we can see degradation in the roofs of the houses and around the windows, where we have edges or sharp transitions.

However, it is not easy to observe degradations in regions with over exposure such as the surface of the lighthouse.

3.1 Visual System-based Representations

Eyes are the connectors between a stimuli and the visual system. Therefore, to understand perceived quality, we need to understand the structure of an eye and its functionailties. We show the basic components of an eye in Fig. 5.

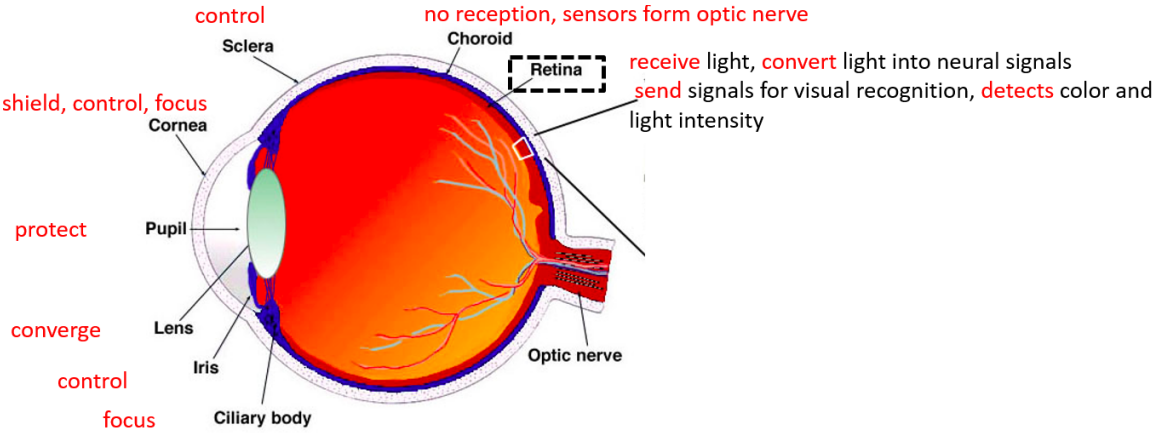


Figure 5: Main sections of an eye. This image is obtained from the study [94].

A sclera, an iris, and a cornea structure are responsible for the control of an incoming light. The cornea structure is also responsible for the shield and the control of incoming light [95]. A pupil is necessary for the protection of an eye, a lens is required for the convergence of incoming light, a ciliary body is used to focus incoming light, and a choroid is responsible for feeding the structures in an eye with oxygen and nutrients. All these structures can be considered as part of an acquisition system. A retina is not only responsible for the acquisition stage, but also for processing. A retina receives incoming light, converts the light into neural signals, sends the neural signals for visual recognition, and detects color and light intensity.

To understand the roles of a retina, we need to look into the main components of a retina, which are shown in Fig. 6. The direction of incoming light is from left to

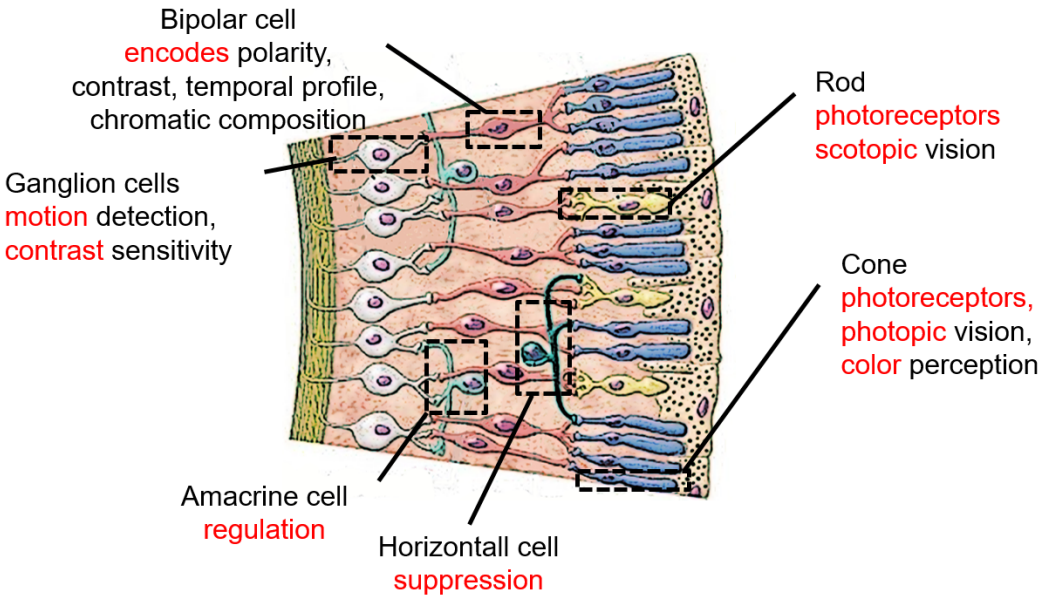


Figure 6: Main structures of a retina.

right and it is received by rods and cones. A rod is a photoreceptor responsible for scotopic vision, which corresponds to the vision of an eye under low light conditions. A cone is a photoreceptor that is responsible for vision under well-lit conditions, which corresponds to photopic vision. In addition to the photopic vision, a cone is also responsible for color perception. These photoreceptors transfer signals to horizontal cells, which are responsible for suppression of signals, and to bipolar cells, which are responsible for encoding the characteristics of received signals. After the horizontal and the bipolar cells, signals are transmitted to ganglion cells, which are responsible for motion detection and contrast sensitivity. Since the scope of this thesis is limited to images, motion detection is not considered. However, contrast sensitivity mechanisms of retinal ganglion cells need to be understood in order to model perception.

3.1.1 Retinal Ganglion Cell-based Difference Map

We can mimic the contrast sensitivity effect in perception by using the contrast sensitivity models of retinal ganglion cells. Based on the observations in [42, 43], a difference of Gaussian (DoG) formulation can be used as a primitive model. Multiple DoG operators can be used to decompose an image into different frequency bands,

which lead to a parameter set of three. Because of the multiple parameter tuning requirement, using DoG operators in a contrast sensitivity model is not feasible. However, a Laplacian of Gaussian operator (LoG) has a single parameter that requires tuning. The LoG operator corresponds to the second derivative of a Gaussian, which can be approximated by DoG operators. Therefore, LoG operators are used to mimic contrast sensitivity in a visual system because of its simplicity and similarity to retinal ganglion cell structures. The formulation of a LoG operator can be given as

$$LoG[m,n] = \frac{1}{\sqrt{2\pi\sigma^2}} \frac{m^2+n^2-2\sigma^2}{\sigma^4} e^{-\frac{m^2+n^2}{2\sigma^2}}, \tag{1}$$

where m and n are the coordinates of the filter with respect to the center and σ is the standard deviation.

We can obtain a numerical approximation of the Laplacian operator using the Taylor series expansion. Let's assume that we have a function $f(x)$ and h is a small increment. We can use the Taylor series expansion to obtain the formulation of $f(x+h)$ as

$$f(x+h) = f(x) + hf'(x) + \frac{1}{2}h^2 f''(x) + \frac{1}{3!}h^3 f'''(x) + O(h^4), \tag{2}$$

and $f(x-h)$ as

$$f(x-h) = f(x) - hf'(x) + \frac{1}{2}h^2 f''(x) - \frac{1}{3!}h^3 f'''(x) + O(h^4), \tag{3}$$

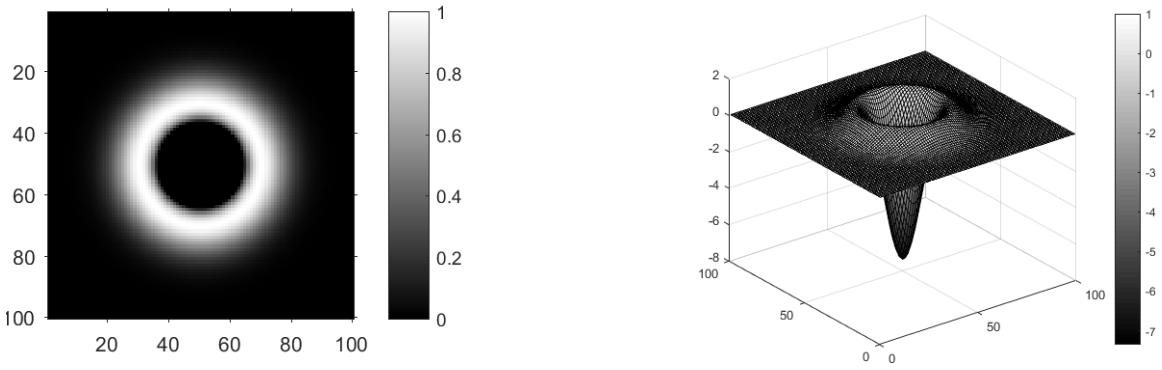
where the $'$ sign corresponds to the first derivative, $''$ to the second derivative, and $'''$ to the third derivative, the ! sign denotes the factorial, and $O()$ is the big O notation. We sum the formulations in Eq. (2) and Eq. (3) to cancel out some of the terms and obtain

$$f(x+h) + f(x-h) = 2f(x) + h^2 f''(x) + O(h^4), \tag{4}$$

where the terms with the first and the third order derivatives cancel each other out. We shift the $2f(x)$ term to the left hand side and divide all the terms by h^2 to obtain

$$\frac{f(x+h) - 2f(x) + f(x-h)}{h^2} = f''(x) + O(h^2), \tag{5}$$

where $O(h^2)$ is negligible compared to $f''(x)$. Thus, a Laplacian operator in one dimension is formulated with the filter coefficients [1 -2 1] in the digital domain. To obtain a two-dimensional Laplacian filter, we sum a horizontal and a vertical Laplacian filter. The summation leads to a 3 by 3 matrix, in which the central element is minus four, elements located at the top, the bottom, the left, and the right are one, and all other elements are zero. We multiply a 3 by 3 numerical approximation of a Gaussian filter with a Laplacian operator to obtain a Laplacian of Gaussian (LoG) filter. To visualize the characteristics of a LoG filter, we show a 100×100 LoG filter with a standard deviation of 10 in two and three dimensional representations in Fig. 7. We use a grayscale color coding in which white leads to high values and black leads to low values.



(a) Magnitude versus 2-D spatial coordinate

(b) 3-D surface

Figure 7: Visualization of the 2-D impulse response of a LoG operator.

The shape of the two dimensional representation of a LoG filter is similar to a blob structure, which can be used to detect changes in the surround with respect to the central region. The representation of the LoG filter in three dimension also shows how the filter can amplify the relative difference between the center and the surround.

To obtain a retinal ganglion cell-based difference map, we convolve both a reference image and a distorted image with a LoG operator and we calculate the absolute difference as

$$RGCD_{i,c} = \left| I^R_{i,c} * LoG - I^D_{i,c} * LoG \right|, \tag{6}$$

where $*$ is the convolution operator, $RGCD$ is the retinal ganglion cell-based difference map, c is the color channel index, and i is the window index, which is a function of m and n in the LoG operator. Because RGCD is calculated pixel-wise rather than block-wise, the window index is equivalent to the pixel index. RGCD is computed for each pixel in every color channel in the RGB space. Then, the three resulting maps are combined pixel-wise using geometric mean as

$$RGCD_i = \sqrt[3]{RGCD_{i,1} \cdot RGCD_{i,2} \cdot RGCD_{i,3}}. \tag{7}$$

The RGCD map corresponding to the images in Fig. 4 is given in Fig. 8. In this visualization, high values correspond to significant degradations and low values indicate minor level degradations. The color of the pixels goes from blue to red under significant degradation as shown in the color bar. Lowly textured regions are smoothed out and highlighted degradations correspond to sharp transitions. RGCD captures most of the degradation around the houses, the lighthouse, and the clouds, and some degradation within the sky and the rocks. Nonetheless, RGCD is not very sensitive to the level of color-based degradation as can be seen in the upper row corresponding to the sky.

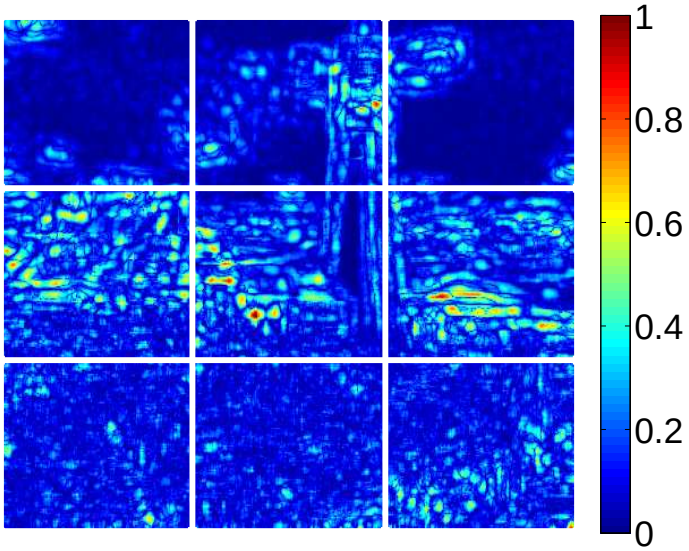


Figure 8: Retinal ganglion cell-based difference (RGCD) map.

3.1.2 Cortical Neuron-based Structural Difference Map

The functional role of neurons and neural systems have been investigated and primtive models have been developed to understand their non-linearities and adaptation mechanisms [37]. These non-linearities in the visual system are reduced by suppression mechanisms, which can be modeled by normalization operations [40]. We model these mechanisms using local normalization as a two-step process. In the first step, the local mean is calculated as

$$\mu_{i,c}^{R} = \frac{1}{W^2} \sum_{m=m_0+1}^{m_0+W} \sum_{n=n_0+1}^{n_0+W} I_c^R[m,n], \tag{8}$$

where m_0 and n_0 are the indices of the top left coordinate of each window, m and n are the indices of the processed pixels, W is the window size, c is the color channel index, i is the window index, which depends on m_0 and n_0, I^R is the reference feature map, and I^D is the distorted feature map. Then, the local mean map (μ^R) is interpolated using a bicubic operation and subtracted from the input image. This operation is performed for each color channel.

In the second step, the local standard deviation is computed by

$$\sigma_{i,c}^{R} = \sqrt{\frac{1}{W^2} \sum_{m=m_0+1}^{m_0+W} \sum_{n=n_0+1}^{n_0+W} (I_c^R[m,n] - \mu_{i,c}^{R})^2)}, \tag{9}$$

where μ^R is the reference mean map, σ^R is the reference standard deviation map, W is the window size, m_0 and n_0 are the indices of the top left coordinate of each window, i is the index of each window, which depends on m_0 and n_0, c is the color channel index, I^R is the reference image, and I^D is the distorted image. The mean shifted values are divided by the local standard deviation ($\sigma_{i,c}^R$) over each color channel for each pixel. The absolute difference between the normalized images is denoted as the structural difference (SD) and is formulated as

$$SD_{i,c} = \left| LN(I_{i,c}^R) - LN(I_{i,c}^D) \right|, \tag{10}$$

where i the window index, c is the color channel index, $|\cdot|$ is the absolute value oper-

ator, and LN corresponds to the cortical neuron-based local normalization operator. Structural difference maps are calculated for each color channel and these maps are pooled pixel-wise using a geometric mean operation which can be formulated as

$$SD_i = \sqrt[3]{SD_{i,1} \cdot SD_{i,2} \cdot SD_{i,3}}. \tag{11}$$

The structural difference map corresponding to the images in Fig. 4 is given in Fig. 9. In this visualization, high values correspond to significant degradations and low values indicate minor level degradations. The color goes from blue to red under significant degradation as shown in the color bar. The block-wise nature of local normalization leads to discontinuities among non-overlapping windows, especially around the window borders. Within each window, there are fluctuations and inconsistencies because of the pixel-wise operations. Sky and cloud regions lead to high SD values whereas textured regions, such as rocks and buildings, have lower SD values. Smooth regions around houses and lighthouse such as walls lead to lower SD values compared to regions with sharp transitions and edges such as regions around windows or edges of the roofs.

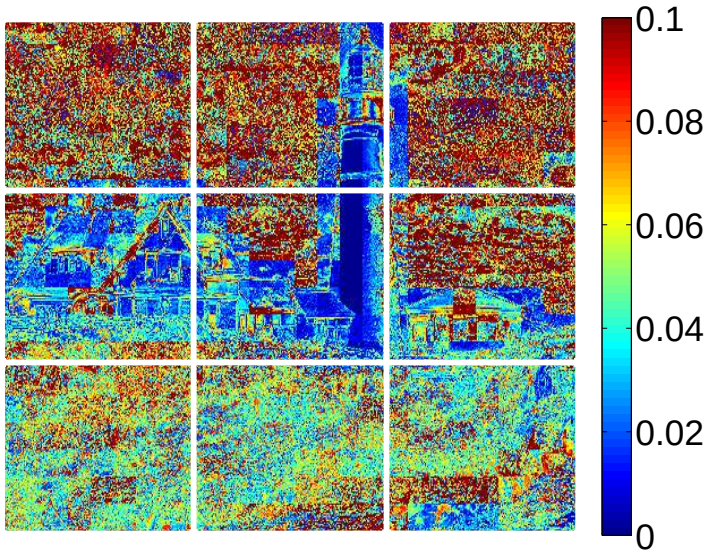


Figure 9: Structural difference (SD) map.

3.2 Color-based Representations

We analyze various approaches to quantify color degradations and a toy example is used to visualize the insights of these methods [96]. In the toy example, we provide a color chart given in Fig. 10, in which we have six different color tones. The first column contains the reference colors, the second column contains the color tones close to the reference colors, and the third column contains the colors that are significantly different from the reference colors. The reference colors are compared with the color tones in the same row. Thus, there are two comparisons in each row and four in total. Each approach will give us a number proportional to the differences between color tones. Ideally, we should compare subjective scores and estimated scores. However, as subjects, it is difficult to numerically state the differences or the similarities between color tones. Even though subjects can not confidently assign a quality score, they can at least state similarities among colors. Therefore, the objective of the toy example is to investigate whether analyzed approaches can differentiate the similarity of color tones.

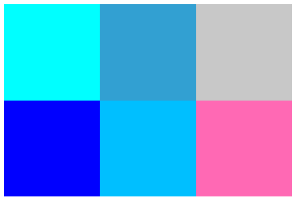


Figure 10: Color chart with six color tones.

3.2.1 Pixel-wise Chroma Fidelity

By default, images are usually represented in the RGB color format. An RGB image is composed of red, green, and blue color channels, in which color and intensity information are mixed. To separate color and intensity information, we can transfer images from the RGB color space to different color spaces. We summarize various color space design approaches in the literature from 16^{th} to 20^{th} century in Fig. 11.

Figure 11: Color space design and modeling approaches in the literature.

Out of all these approaches, CIEXYZ corresponds to one of the first attempts to produce a color space based on measurements of human color perception. CIEXYZ is further extended with more subjective formulations to obtain CIELa*b*, which is a perceptually uniform and a device independent color space. Perceptual uniformity means that numerical differences in a color space correlate to perceived differences. Therefore, instead of obtaining pixel-wise differences in the RGB color space, we transform images into the La*b* color space, in which luma and chroma information are separated, and calculate an absolute difference between the chroma channels of compared images. The color tones in the toy example are shown in Fig. 12, in which pixel-wise differences in chroma channels a* and b* are shown on top of the color tones. The difference based on a* channel on the top row and the difference based on b* channel on the bottom row detect the relatively close colors. However, both chroma-based differences fail in the opposite row.

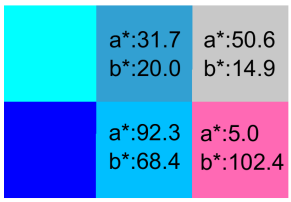


Figure 12: Chroma fidelity chart.

3.2.2 Color Difference Equations

Instead of directly calculating pixel-wise differences in chroma channels, we can use color difference equations that are specifically designed for color tone matching in the color science community. The International Commission on Illumination (CIE) determines the lighting-related standards including color difference equations. One of the most commonly used color difference equation designed by CIE is CIEDE2000 [30, 31]. Even CIEDE2000 is originally used for tone matching applications, the application area of CIEDE2000 is also extended to image quality assessment [26].

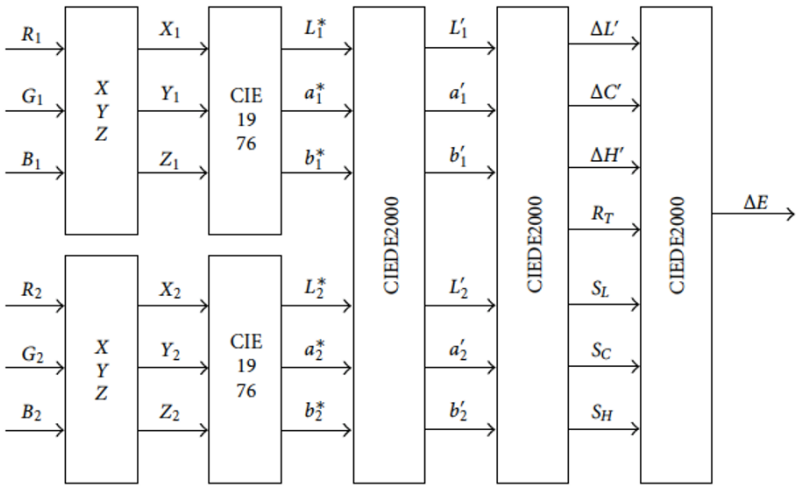


Figure 13: CIEDE2000 pipeline. This image is obtained from the study [26].

The CIEDE2000 [26] color difference equation consists of lightness difference, chroma difference, and hue difference equations. CIEDE2000 can be formulated as

$$CIEDE_i = \sqrt{\left(\frac{\mu^R_{i,1} - \mu^D_{i,1}}{K_L S_L}\right)^2 + \left(\frac{\mu^R_{i,2} - \mu^D_{i,2}}{K_C S_C}\right)^2 + \left(\frac{\mu^R_{i,3} - \mu^D_{i,3}}{K_H S_H}\right)^2 + R_T \cdot \left(\frac{\mu^R_{i,2} - \mu^D_{i,2}}{K_C S_C}\right) \cdot \left(\frac{\mu^R_{i,3} - \mu^D_{i,3}}{K_H S_H}\right)}, \quad (12)$$

where i is the window index, K_L, K_C, and K_H are the environmental tuning parameters, S_L, S_C, and S_H are the weighting factors that are functions of l, $a*$ or $b*$ color channel values, R_T is the rotation factor, which is a non-linear function of chroma channels included for color correction. The CIEDE2000 pipeline is summarized in Fig. 13. A detailed description of the rotation and the weighting factors can be found in [26]. The CIEDE2000 differences are shown on top of the color tones in Fig 14. The color difference between similar tones are higher than the difference between less similar tones. Therefore, CIEDE2000 difference fails in differentiating color tones in the toy example.

Figure 14: CIEDE2000 color difference chart.

3.2.3 Color Name Distance

Color difference equations consider the perceptibility of colors using subjective data in the formulation process. However, color difference formulations do not work well under significant color distortions because these formulations are designed to compare similar color tones (within 7 CIELab unit). Therefore, we use color name distances to measure significant color degradations.

3.2.3.1 Color Names

Color names are pixel-based descriptors in which each entry corresponds to the probability of that pixel being perceived as one of the N basic colors. In the case of $N = 11$,

Figure 15: Color name descriptors.

the color names in the dictionary are: black, blue, brown, grey, green, orange, pink, purple, red, white, and yellow [97]. Color name descriptors are introduced in [98], in which images are searched online using color terms and a color name lookup table is obtained based on the color values in hand-segmented images. For example, if we have images that are labeled as red and green, their pixels will be mapped to the La*b* color space representing their labels as shown in Fig. 16 . In the following sections, we denote $L(\cdot)$ as the color name lookup operator that receives RGB values as input, transforms them into the La*b* domain, and returns N dimensional color name probability vectors.

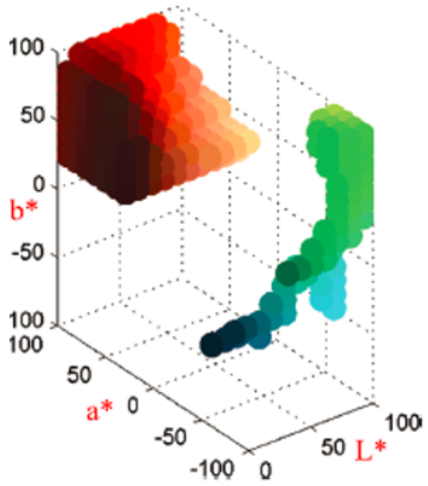


Figure 16: A toy example that shows the distribution of colors in the La*b* color space. This image is obtained from the study [98].

3.2.3.2 Distance Measures

To measure the difference between color name descriptors, we can use l-norms. However, l-norms do not necessarily measure the perceptual difference. We introduce a toy example to analyze whether difference or distance methods capture the perceptually similar histogram pairs or not as in [99]. Normalized histograms are shown in Fig. 17. We use l_1 norm as a bin-by-bin dissimilarity measure, quadratic form as a cross-bin dissimilarity measure, and Earth Mover's Distance (EMD) as an alternative to other measures.

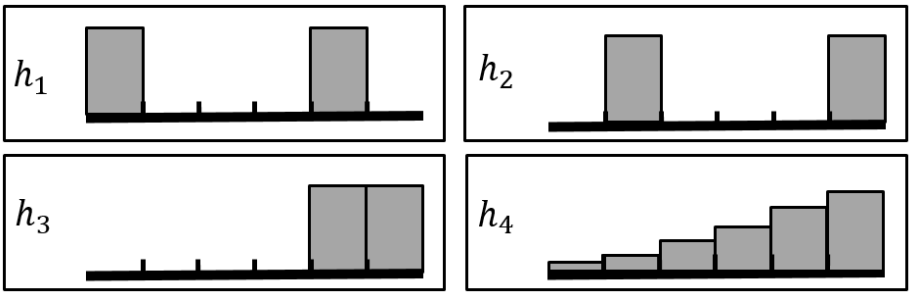


Figure 17: Perceptual difference toy example: Distributions.

In the first scenario, we compare the relative difference between the h_1-h_2 pair and the h_1-h_3 pair. Perceptually, h_2 is more similar to h_1 because it is only a shifted version. Bin-by-bin methods compare the histograms as shown in Fig. 18.

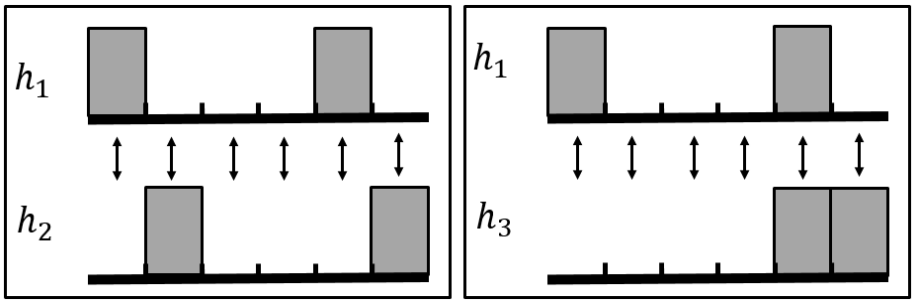


Figure 18: Perceptual difference toy example: Scenario 1 - Bin by bin dissimilarity.

When we use an l_1 norm to measure the difference between h_1 and h_2, it sums all the elements in both histograms because none of them are in the same location. However, in the case of h_1-h_3 comparison, one of the entities are in the same location, which decreases the sum in the l_1 norm calculation. On the contrary of a subjective opinion, l_1 norm considers the h_1-h_3 pair more similar than the h_1-h_2 pair. Instead of using a bin-by-bin dissimilarity, we can also use a cross-bin dissimilarity, which is shown in Fig. 19.

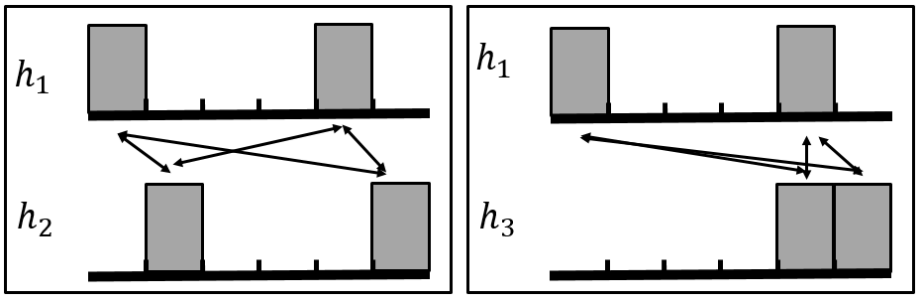


Figure 19: Perceptual difference toy example: Scenario 1 - Cross-bin dissimilarity.

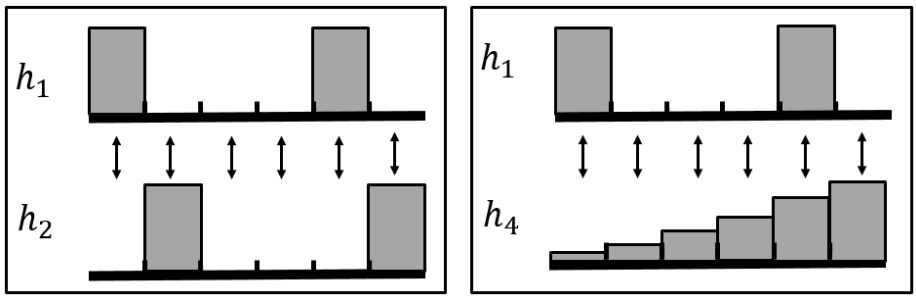
As a cross-bin dissimilarity measure, we use the quadratic form distance, which is formulated as

$$Q(h_1, h_2) = (h_1 - h_2)^T A (h_1 - h_2), \tag{13}$$

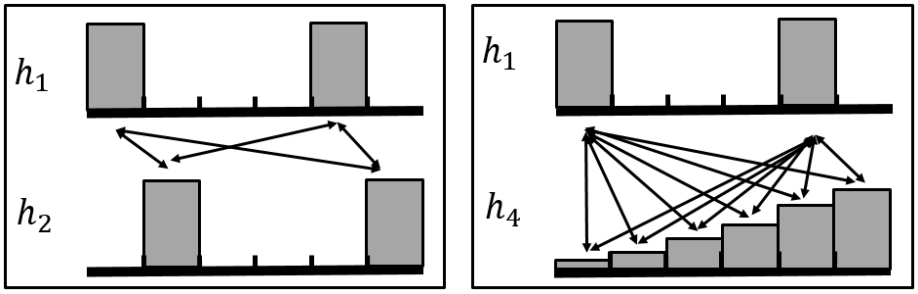
where T is the transpose operation and A is the similarity matrix. The quadratic form distance is 0.44 for the h_1-h_2 pair and it is 0.70 for the h_1-h_3 pair. Therefore, according to the quadratic form distance, the h_1-h_2 pair is more similar than the h_1-h_3 pair, which is same as the subjective opinion. Therefore, in the first scenario of the toy example, l_1 norm fails whereas quadratic form distance leads to perceptually correlated measurements.

In the second scenario, we compare the h_1-h_2 pair to the h_1-h_4 pair. The bin-by-bin and the cross-bin dissimilarities are shown in Fig. 20. Bin-by-bin dissimilarity measured by the l_1 norm is 2.00 for the h_1-h_2 pair and it is 1.42 for the h_1-h_4 pair.

Cross-bin dissimilarity measured by the quadratic form is 0.44 for the h_1-h_2 pair and it is 0.42 for the h_1-h_4 pair. According to the l_1 norm and the quadratic form, the h_1-h_4 pair is more similar than the h_1-h_2 pair, which contradicts with the subjective opinion. Even the quadratic form works for the first scenario, it fails for the second scenario.



(a)Bin-by-bin dissimilarity.



(b) Cross-bin dissimilarity

Figure 20: Perceptual difference toy example: Scenario 2.

Neither the l_1 norm nor the quadratic form works for both of the scenarios. Therefore, we use an alternative approach, which is Earth Mover's Distance (EMD). EMD is a type of cross-bin dissimilarity so symbolic relationship among bins are same as Fig. 19 and Fig. 20(b). EMD for the h_1-h_2 pair is 0.20 and it is 0.50 for the h_1-h_3 pair. Based on EMD, the h_1-h_2 pair is more similar than the h_1-h_3 pair. EMD for the h_1-h_4 pair is 0.26, which implies that the h_1-h_2 pair is more similar than the h_1-h_4 pair. In both of the scenarios in the toy example, the pairs detected by EMD as more similar are same as the subjective opinion.

3.2.3.3 Earth Mover's Distance

Earth Mover's Distance (EMD) is based on a solution approach to the transportation problem in linear optimization. The intuition behind EMD is to calculate the minimum cost required to transform one distribution into the other [99]. The EMD operator takes into account all the flow scenarios between two distributions to obtain the distance [96] as

$$EMD(x, y) = \min_{f_{k,l}} \left\{ \sum_{k=1}^{N} \sum_{l=1}^{N} d_{k,l} f_{k,l} \right\}, \tag{14}$$

where x and y are the compared distributions, k is the index of the entities in x, l is the index of the entities in y, N is the number of entities in x and y, $f_{k,l}$ is the flow from the k^{th} entitiy in x to the l^{th} entity in y, and $d_{k,l}$ is the ground distance between the compared entities. The constraints of the flow equation [96] are formulated as

$$\sum_{k=1}^{N} \sum_{l=1}^{N} f_{k,l} = 1, \quad f_{k,l} \geqslant 0, \tag{15}$$

where sum of the overall flow adds up to unity and the flow is defined to be non-negative. The ground distance between the entities can be defined based on the application.

3.2.3.4 Measuring the Color Name Distance

The color name distance between two pixels is given by

$$CND_i = EMD(L(\mu_i^R), L(\mu_i^D)), \tag{16}$$

where μ^R is the locally mean pooled reference map, μ^D is the locally mean pooled distorted map, i is the window index, and `EMD` is the Earth Mover's Distance operator. We can express Eq. (14) in Section 3.2.3.3 as

$$EMD(L(\mu_i^R), L(\mu_i^D)) = \min_{f_{k,l}} \left\{ \sum_{k=1}^{N} \sum_{l=1}^{N} d_{i,k,l} f_{i,k,l} \right\}, \tag{17}$$

where i is the window index, k is the index of the entity in the reference color name descriptor, l is the index of the entity in the compared color name descriptor, N is

the number of entities in the color descriptor or in other words the number of colors in the dictionary, $f_{i,k,l}$ is the flow from the k^{th} color probability in the reference to the l^{th} color probability in the compared descriptor for the i^{th} window, and $d_{i,k,l}$ is the ground distance between color names. The probability of a pixel being affiliated to one of the color names is obtained by summing up the flow vectors leading to an indexed color name [96] given by

$$P_{i,l} = \sum_{k=1}^{N} f_{i,k,l}, \tag{18}$$

where $P_{i,l}$ is the l^{th} entity in the color label descriptor for i^{th} window and the full descriptor is expressed as $[P_{i,1}, P_{i,2}, ..., P_{i,N}]$. The constraints of the flow equation [96] are formulated as

$$\sum_{k=1}^{N}\sum_{l=1}^{N} f_{i,k,l} = 1, \quad f_{i,k,l} \geqslant 0, \tag{19}$$

where sum of the overall flow adds up to unity and the flow is defined to be non-negative.

The entities in color name descriptors correspond to different colors and perceived differences between different colors are not same. Therefore, instead of using a uniform distance, the flow between entities in color name descriptors is weighted based on perceived color differences. The joint distribution of basic color terms in the La*b* color space is used to obtain the perceived distance between colors [96] and these ground distances between basic color tones are given in Fig. 21. If two color tones are same, ground distance is 0.00. When color tones are perceptually similar as shown in Fig. 22, such as black-brown, black-grey, brown-grey, grey-white, and orange-red, ground distance values are in between 0.68 and 0.94. The ground distance between all other colors is 1.00.

When the ground distance is 1.00, we can directly use the flow in Eq. (14) to calculate the EMD distance. In the case of significant color degradations, the ground distance between the entities in the color name descriptors increases the total cost of

	Black	Blue	Brown	Grey	Green	Orange	Pink	Purple	Red	White	Yellow
Black	0	1	0.94	0.76	1	1	1	1	1	1	1
Blue	1	0	1	1	1	1	1	1	1	1	1
Brown	0.94	1.00	0	0.93	1	1	1	1	1	1	1
Grey	0.76	1	0.93	0	1	1	1	1	1	0.68	1
Green	1	1	1	1	0	1	1	1	1	1	1
Orange	1	1	1	1	1	0	1	1	0.92	1	1
Pink	1	1	1	1	1	1	0	1	1	1	1
Purple	1	1	1	1	1	1	1	0	1	1	1
Red	1	1	1	1	1	0.92	1	1	0	1	1
White	1	1	1	0.68	1	1	1	1	1	0	1
Yellow	1	1	1	1	1	1	1	1	1	1	0

Figure 21: Ground distances between different color tones.

Figure 22: Similar color tones.

transformation in the EMD distance.

3.2.3.5 CND Between Color Tones and Compared Images

The CND distances are shown on top of the color tones in Fig. 23. Similar color tones lead to lower CND distance, which shows that CND can successfully differentiate the color tones in the toy example.

Figure 23: CND distance chart.

The CND map corresponding to the images in Fig. 4 is given in Fig. 24, which

is resized to the original image resolution using the bicubic interpolation. In this visualization, high values correspond to significant degradations and low values indicate minor level degradations. The color goes from blue to red under significant degradations as shown in the color bar. CND detects the degradations around the sky and the clouds as the most degraded part. Distortion levels of textured regions are detected as either low or mediocre.

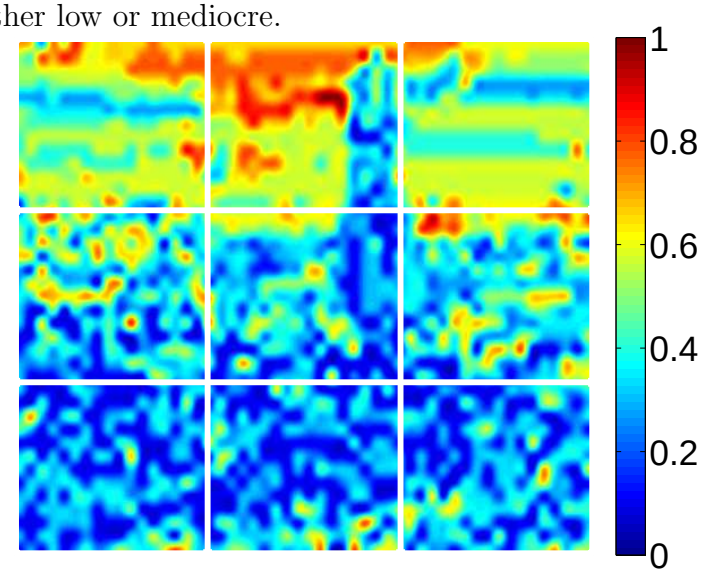


Figure 24: Color name distance (CND) map.

3.3 Visual System- and Color-based Representations

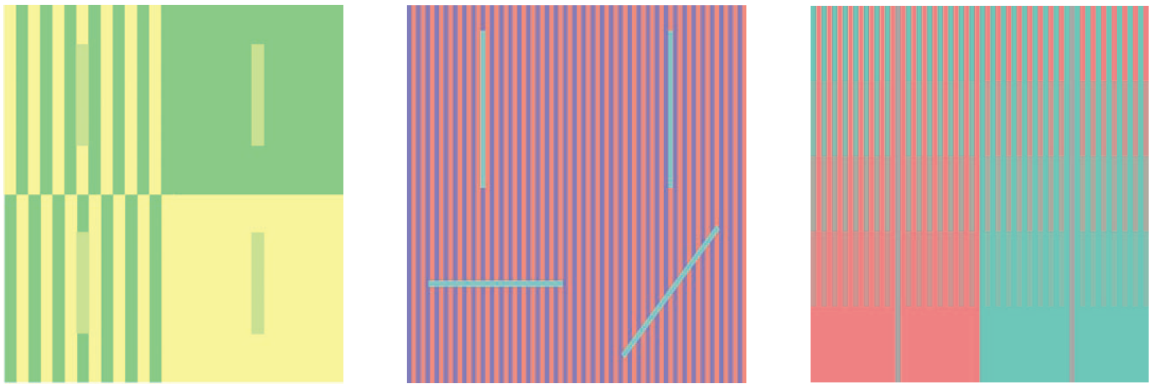
In Section 3.2, the methods that we discuss are pixel-wise methods. However, a visual system perceives structures rather than pixels. Therefore, in this section, we describe a visual system model that considers center-surround effects.

3.3.1 Chromatic Induction Model

The authors in [100] introduced a brightness-based low-level induction model (BIWaM) using multiresolution wavelets. BIWaM was shown to mimic basic perception mechanisms including but not limited to simultaneous contrast, the White effect, grating induction, the Todorovic effect, Mach bands, the Chevreul effect, the Adelson-Logvinenko tile effects, and the dungeon illusion. BIWaM also unified the brightness

contrast and assimilation effects into a single model. Brightness contrast describes the phenomenon when the brightness of a test stimuli shifts away from the surroundings and brightness assimilation is the opposite case when the shift is toward the surroundings.

Chromatic induction model (CIWaM), which is an extension of brightness model, mimics the chromatically opponent visual pathways. CIWaM model is based on three main observations. Spatial frequency effect is the first observation, which states that the perception of the central stimuli is influenced by the frequency characteristics of the surround stimuli. The second observation is spatial orientation effect, which means that the similarity between the orientation of central and surround stimuli leads to assimilation of the central stimuli whereas difference in the orientation leads to contrast. Finally, the third observation is surround contrast effect, which indicates that the contrast of the surrounding stimuli leads to assimilation of the central stimuli. A toy example for each observation is given in Fig. 25.



(a) Spatial frequency (b) Spatial orientation (c) Surround contrast

Figure 25: Toy examples that show the center-surround effects. This image is obtained from the study [101].

3.3.2 Spatiochroamtic Grouping Map

The authors in [101] extended the chromatic induction model with low-level spatiochromatic grouping to estimate saliency, which is used as a visual quality assistance map in this work.

3.3.2.1 Introduction to Spatiochromatic Grouping

Images are frequently transformed from the RGB color space to opponent color spaces. For each color channel, we perform following operations. First, wavelet transform is applied to obtain wavelet planes. Then, grouplet transform is applied over the wavelet planes. Center contrast normalization and contrast sensitivity adjustment follow the grouplet transform. Bicubic interpolation is used to obtain the original resolution and inverse wavelet transform is applied to go back to spatial domain. Euclidean norm is used to obtain spatiochromatic grouping map as shown in Fig. 26. In terms of notation, instead of using a single pixel index, we use two indices to describe grouplet transform.

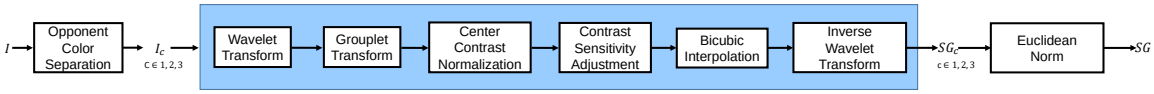


Figure 26: Spatiochromatic grouping block diagram.

3.3.2.2 Color Space Transformation

At first, an RGB image is transformed to an opponent color space, after gamma (γ) correction, as

$$I_1 = \frac{R-G}{R+G+B}, I_2 = \frac{R+G-2B}{R+G+B}, I_3 = R+G+B, \tag{20}$$

where I_1, I_2, and I_3 are the opponent color channels and R, G, and B are the RGB channels. Images are transformed from RGB to opponent color space to separate luminance and chrominance information. The sensitivity of a visual system is different

for luminance and chrominance components. Therefore, once these components are separated, different contrast sensitivity formulations can be used to model visual system characteristics, which are explained in 3.3.2.6.

3.3.2.3 Spatial Decomposition

Spatial decomposition is performed over each color channel map to obtain scale and orientation information as

$$\{w_{s,o}\}_{1\leqslant s\leqslant S, o=h,v,d}\,, \tag{21}$$

where $w_{s,o}$ is the wavelet plane at spatial scale s and orientation o, horizontal, vertical, and diagonal orientations are represented with h, v, and d. Gabor-like basis functions are used in the wavelet transform to mimic the receptive fields of neurons in a visual cortex.

3.3.2.4 Grouplet Transform

We use the grouplet transform to enhance abstract representations and suppress non-salient features as introduced in [101]. At first, the approximation component (low frequency) of the wavelet plane is initialized as $a_{s,1,o}$ and other scales can be obtained as

$$a_{s,j+1,o}\left[m,n\right] = \frac{a_{s,j,o}\left[2m-1,n\right] + a_{s,j,o}\left[2m,n\right]}{2}, \tag{22}$$

where m and n are the pixel indices, a is the approximation (low frequency) component, and j is the scale. The detail component (high frequency) is calculated as follows

$$d_{s,j+1,o}\left[m,n\right] = \frac{a_{s,j,o}\left[2m,n\right] - a_{s,j,o}\left[2m-1,n\right]}{2^{j}}, \tag{23}$$

where d is the normalized difference of the consecutive approximation components at grouplet scale j.

In the Haar transform, the approximation and the detail coefficients are computed between pairs of consecutive elements. Grouplet transform is also a type of

Haar transform but the pairs are not necessarily consecutive. Grouplet transform coefficients are paired along the contour that is common to these coefficients. We can consider contour-based pairing as finding the points in the direction of maximum regularity. Grouplet plane is obtained by computing the detail components $d_{s,j,o}$ for each scale, which can be considered as a sparse representation of complex geometrical structures.

3.3.2.5 Surround Contrast Model

We perform divisive normalization to partially model center-surround contrast mechanism as

$$z_{s,j,o}[m,n] = \frac{(d_{s,j,o}[m,n]^{cen})^2}{(d_{s,j,o}[m,n]^{cen})^2 + (d_{s,j,o}[m,n]^{sur})^2}, \tag{24}$$

where m and n are the pixel indices, d^{cen} is the wavelet coefficient of the central region, d^{sur} is the wavelet coefficient of the surround region, and z is the normalized center contrast. This mechanism is used to formulate surround contrast effect.

3.3.2.6 Contrast Sensitivity Adjustment

We use an extended contrast sensitivity function (ECSF) [101] to model spatial frequency and spatial orientation effects. Normalized coefficients along with spatial frequency and orientation information are used as the input of ECSF as

$$\alpha_{s,j,o}[m,n] = ECSF(z_{s,j,o}[m,n]), \tag{25}$$

where $ECSF$ is the extended contrast sensitivity function and α is the contrast-adjusted and divisive-normalized coefficient. ECSF is defined as the summation of two terms as

$$ECSF(z_{s,j,o}[m,n]) = z_{s,j,o}[m,n] \cdot g(s) + k(s), \tag{26}$$

where the first term is the multiplication of a normalized coefficient and an approximated psychophysical contrast sensitivity function, and the second term is introduced

to set the lower bound non-zero. Psychophysical contrast sensitivity function is expressed as

$$g(s) = \begin{cases} \beta e^{-\frac{(s-s_0^g)^2}{2\sigma_1^2}}, & s \leqslant s_0^g \\ \beta e^{-\frac{(s-s_0^g)^2}{2\sigma_2^2}}, & otherwise \end{cases} \tag{27}$$

where s is the spatial scale, β is the scaling constant, σ_1 and σ_2 are the deviation parameters that formulate the spread of the spatial sensitivity, and s_0^g is the peak scale sensitivity. The second term in the extended contrast sensitivity formulation is expressed as

$$k(s) = \begin{cases} e^{-\frac{(s-s_0^k)^2}{2\sigma_3^2}}, & s \leqslant s_0^k \\ 1, & otherwise \end{cases} \tag{28}$$

where σ_3 is the deviation parameter that formulates the spatial sensitivity of $k(s)$ and s_0^k is the peak scale sensitivity.

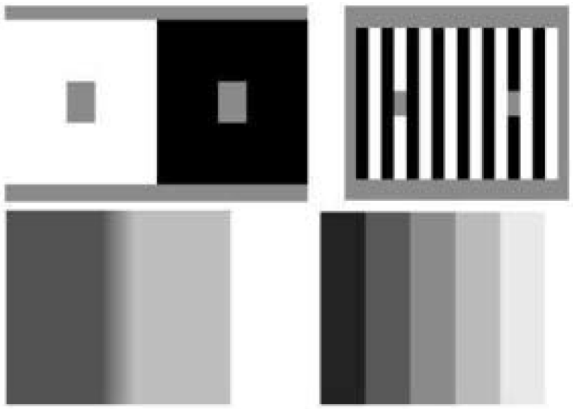


Figure 27: Examples of experimental stimuli for brightness induction. This image is obtained from the study [102].

The parameters of the contrast sensitivity formulations are tuned based on two subjective experiments reported in [102, 103]. The first experiment measured brightness induction in which observers viewed two stimuli with same luminance and different brightness [102]. Observers were asked to adjust the brightness of one of the stimuli to match the brightness of the second stimuli. Examples of experimental stimuli for brightness induction are shown in Fig. 27. The second experiment measured

color induction [103]. Observers performed asymmetric color and brightness matching assignments. Examples of experimental stimuli for color induction are shown in Fig. 28.

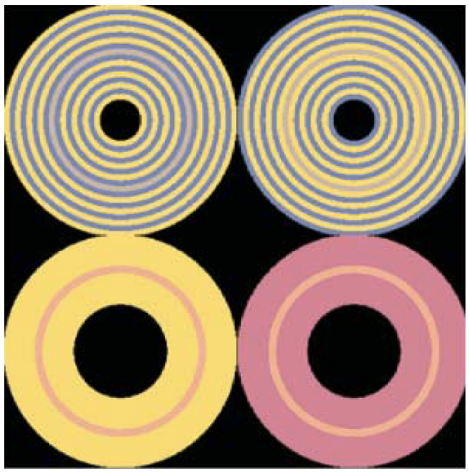


Figure 28: Examples of experimental stimuli for color induction. This image is obtained from the study [103].

Based on the experimental data, least squared regression was used to determine the parameters of the extended contrast sensitivity functions, which are summarized in Table 6. The sensitivity of a visual system with respect to intensity and contrast are different. Therefore, a different parameter set was obtained for both brightness induction experiment and color induction experiment. The characteristic curves of extended contrast sensitivity functions are given in Fig. 29. As observed in the figure, signals are enhanced in a narrow passband and suppressed for low spatial scales. Peak spatial scales in the wavelet domain correspond to peak spatial frequencies between 2 and 5 cycles, which are also supported by the experimental studies in [104]. The outputs of these extended contrast sensitivity functions are used to weight wavelet coefficients.

Table 6: The parameters of the extended contrast sensitivity functions.

Parameters	σ_1	σ_2	σ_3	β	s_0^g	s_0^k
Intensity	1.021	1,048	0.212	4.982	4.000	4.531
Color	1.361	0.796	0.349	3.612	4.724	5.059

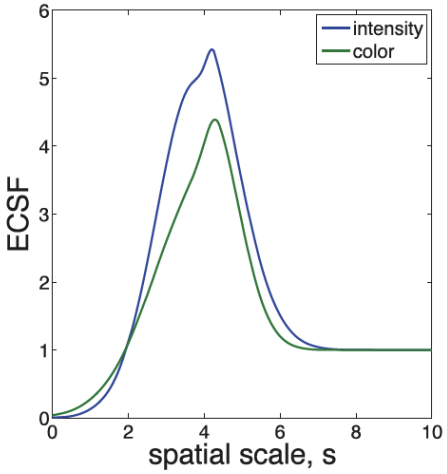


Figure 29: The characteristic curves of extended contrast sensitivity functions. This image is obtained from the study [101].

3.3.2.7 Interpolation and Inverse Transform

Bicubic interpolation is used to resize each plane ($\alpha_{s,j,o}$) and then these planes are summed ($\alpha_{s,o}$). Inverse wavelet is used to transform the wavelet coefficients back to the spatial domain (SG_c), in which c corresponds to the channel index.

3.3.2.8 Color Channel Fusion

Color channels are combined as

$$SG = \sqrt{\sum_{i=1}^{3} (SG_c)^2}, \tag{29}$$

where SG is the spatiochromatic grouping map.

3.3.2.9 Visualization of Spatiochromatic Grouping Map

Spatiochromatic grouping maps corresponding to the images in Fig. 4 are computed and the similarity map between these feature maps is given in Fig. 30. In this visualization, high values correspond to minor level degradations and low values indicate significant degradations. The color goes from red to blue under significant degradation as shown in the color bar. Spatichromatic grouping-based similarity detects the degradations around the sky as the most degraded part. However, as observed in the top middle grid, the regions close to the light tower lead to high similarity values because of the center surround affect. Distortion levels of textured regions are detected as either low or mediocre.

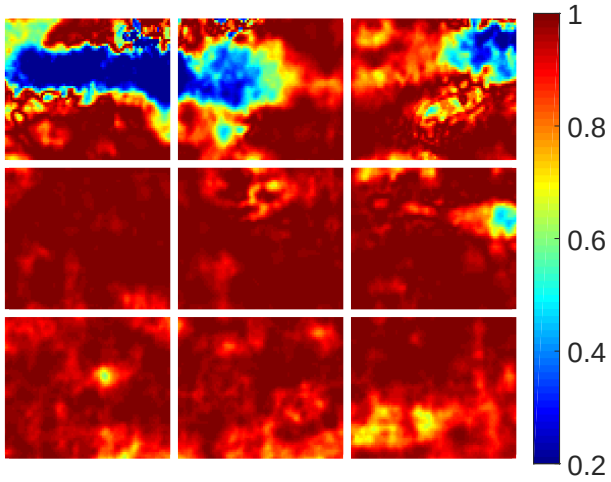


Figure 30: Spatiochromatic grouping-based similarity map.

3.4 Summary

We use a reference image and a distorted image as a toy example to analyze perceived degradations. To capture these perceived degradations, we analyze methods based on visual system characteristics, color measurements, and perception. Specifically, retinal ganglion cell-based difference is used to mimic the effect of contrast

sensitivity, which is formulated by Laplacian of Gaussian operators. Cortical neuron-based structural difference is used to partially model suppression mechanisms, which is formulated by normalization operations.

We use a color chart as a toy example to visualize the insights of alternative color difference measurement techniques including pixel-wise chroma fidelity and color difference equations. We show that methods based on pixel-wise fidelity and color differences do not perceptually measure the difference between colors under significant tone changes. To complement existing methods, we propose measuring distances between color names. To measure the distance between color names, we use l_1 norm as a bin-by-bin dissimilarity measure, quadratic form as a cross-bin dissimilarity measure, and Earth Mover's Distance (EMD). Based on the promising results, we use EMD to measure the difference between color names.

The methods that we discuss in Section 3.2 are pixel-wise methods. However, a visual system perceives structures rather than pixels. Therefore, we use a visual system-based model that considers the center-surround effects. At first, we transform images form the RGB space to an opponent color space. For each color channel, wavelet transform is applied to obtain wavelet planes, grouplet transform is applied over the wavelet planes followed by center contrast normalization and contrast sensitivity adjustment. Normalized and adjusted maps are interpolated back to the original resolution, inverse wavelet transform is applied to go back to spatial domain, and Euclidean norm is used to obtain the final spatiochromatic grouping map.

Throughout this chapter, we describe representations based on visual system characteristics and color perception. Each visual representation captures degradations in a different manner. Therefore, we need to combine these representations considering their characteristics to obtain a hybrid visual representation.

CHAPTER IV

VALIDATION OF IMAGE QUALITY ASSESSMENT

In Chapter 3, we describe visual representations that can capture degradations. We cannot directly measure the accuracy of these visual representations because it is not possible to view the representations inside a visual system. To validate the performance of these representations, we need to pool them to a final quality score and use image quality databases, which include reference and degraded images with corresponding mean opinion scores (MOS). MOS scores of processed images are subtracted from the MOS scores of reference images to obtain DMOS scores. The scores of a high quality image correspond to a high MOS value and a low DMOS value. These scores are obtained from subjective tests, in which users view these images and assign a score based on perceived quality in a controlled setup as shown in Fig. 31. General viewing conditions including the size of the display, the lighting conditions, and the distance with respect to the display, which are shown in Fig. 31, are configured based on the recommendations in Rec. ITU-R BT.500-13 [105].

4.1 Databases

In the validation of the image quality estimators, we use the databases LIVE [106], Multiply Distorted LIVE [107], and TID 2013 [93].

4.1.1 The LIVE Database

The LIVE image quality database contains 29 reference images that are 24-bits/pixel RGB color images with a typical resolution of 768x512 [106]. Reference images are shown in Fig. 32. Each image is distorted with various degradation types and levels to cover the entire perceptual quality range. Distortion types include the JPEG and

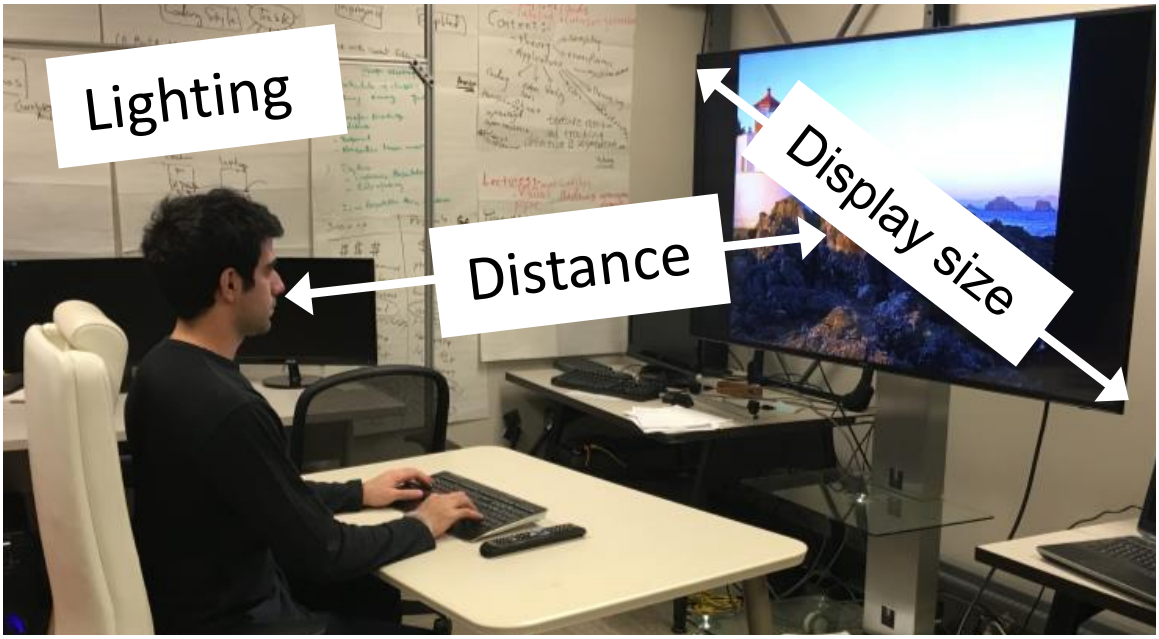


Figure 31: Subjective test setup.

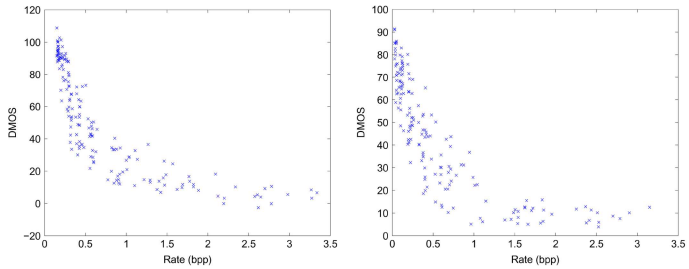
the JPEG2000 compression, white noise in the RGB components, Gaussian blur in the RGB components, and bit errors in the JPEG2000 bitstream when transmitted over a simulated fast-fading Rayleigh channel. In the JPEG2000 compression case, bit rates range from 0.028 bits per pixel (bpp) to 3.15 bpp. In the JPEG compression case, bit rates range from 0.15 bpp to 3.34 bpp. In the case of white noise, standard deviations of the noise range from 0.012 to 2.0 in all the channels of RGB images. In the case of Gaussian blur, standard deviations range from 0.42 to 15 pixels in all the channels of RGB images. In the case of fast-fading errors, the received SNR was varied to generate corrupted bitstreams, whose SNRs range from 15.5 to 26.1.

Observers were asked to score images on a continuous linear scale, which was divided into five equal sections labeled with "bad", "poor", "fair", "good", and "excellent". The number of subjects scoring each image is between 20 and 29. Different experiments with different subjects were conducted using the same equipment and the same viewing conditions to obtain scores for each degradation type. In total, 779 distorted images were evaluated by subjects. In addition, 203 reference images were

Figure 32: Reference images in the LIVE database.

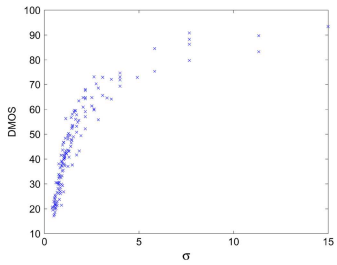
also included in the subjective test. The scores of the reference and the distorted images were used to obtain the difference, which was shifted and scaled to cover the full quality range from 1 to 100. Finally, the mean of the scaled and shifted differences were computed to obtain difference mean opinion score (DMOS). We show the DMOS values versus the distortion levels for each distortion type in Fig. 33. Gaussian blur and white noise follow a monotonically increasing behavior with low standard deviation whereas JPEG and JPEG2000 follow a monotonically increasing behavior with higher deviations. DMOS values in the case of fast-fading distortions are more spread out and they do not follow a clear monotonic curve as in other distortion categories. The distribution of the DMOS scores are shown in Fig. 34. The majority of the scores are distributed between 20 and 90. There are only few instances, in which DMOS scores are higher than 90. We observe an accumulation close to low DMOS scores, which correspond to images with high perceived quality.

A web-based interface, which shows an image and a Java scale-and-slider applet, was used for subjective score entry. Subjective tests were conducted in an office environment with normal indoor illumination levels. The displays used in the tests were 21-inch CRT monitors with a resolution of 1024x768 and same display settings. The

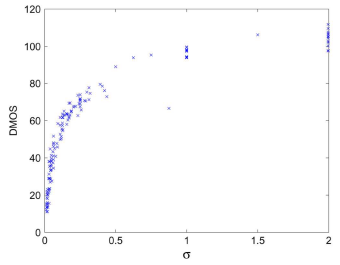


(a) JPEG

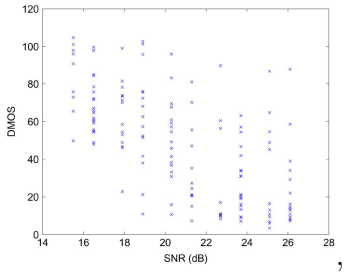
(b) JPEG2000



(c) Blur



(d) Noise



(e) Fast-fading

Figure 33: DMOS versus distortion level in the LIVE database. This image is obtained from the study [106].

distance between a subject and a display was around 2-2.5 times screen heights. Subjective experiments were conducted in multiple sessions: white noise in one session, Gaussian blur in one, channel errors in one, JPEG in two, and JPEG2000 in two sessions. Because of the session time recommendation and the number of images, multiple sessions were required. The scores from different sessions were aggregated through scale realignment to obtain a single score set. The number of images and the number of subjects in each session are summarized in Table 7. The participants in the subjective tests were mostly male undergraduate or graduate students, who were

not experienced with the details of image impairments or quality assessment.

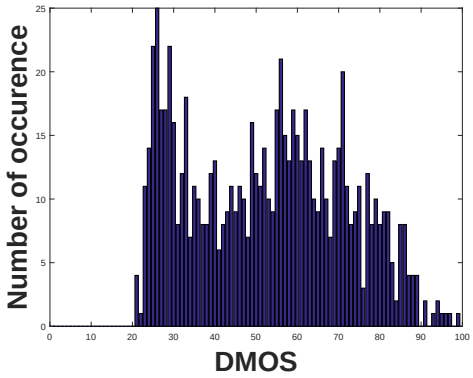


Figure 34: Normalized histogram of subjective scores in the LIVE database.

Table 7: The number of images and subjects in each test session in the LIVE database.

Session	Number of images	Number of subjects
JPEG2000 session 1	116	29
JPEG2000 session 2	111	25
JPEG session 1	116	20
JPEG session 2	117	20
White noise	174	23
Gaussian blur	174	24
Fast-fading	174	20
Total	982	22.8 (average)
Alignment study	50	32

4.1.2 The Multiply Distorted LIVE Database

Even though the LIVE database contains various distortion types, the distortions were not applied simultaneously. In contrast, the multiply distorted LIVE (MULTI) database [107] contains reference images that were degraded with multiple distortions simultaneously. The subjective study was composed of two main parts. In the first part, image storage artifacts were modeled by blurring and then compression with JPEG. In the second part, image acquisition artifacts were modeled by blurring and

then corrupting with white Gaussian noise. Blurring was used to account for narrow depth field and white noise for sensor noise. Gaussian kernels were used for blurring with a square kernel window of three standard deviation over each color channel in RGB images. Standard deviation ranged from 3.2 to 4.6 with a step size of 0.7. Quantization parameters for the JPEG compression were 12, 18, and 27. Noise was generated from a standard normal pdf, whose standard deviations were 0.002, 0.008, and 0.032. In each part of the subjective study, 15 reference images were used to generate 225 images, in which 90 images were singly distorted and 135 images were multiply distorted.

Images were displayed on an LCD monitor at a resolution of 1280×720 at 73.4 ppi. Display settings were configured based on the suggestions in [108] and the study was conducted in a normal workspace environment under standard illumination levels. The distance between a subject and the display was around 4 times the diagonal screen size. The MATLAB Psychometric Toolbox [109] was used to display images and to obtain subjective scores. Images were displayed for 8 seconds and then users were asked to score the image with a slider, which ranged from 0 to 100 and contained semantic labels including "bad", "poor", "fair", "good", and "excellent".

The participants in the subjective tests were mostly male graduate students. The number of participants was 19 in the first part of the experiment and 18 in the second part of the experiment. The scores of the reference and the distorted images were used to obtain the differences and the average of the differences were calculated to obtain the difference mean opinion scores (DMOS). The distribution of the DMOS scores are shown in Fig. 36. In the LIVE database, there is an accumulation close to low DMOS scores. On contrary, in the MULTI database, we observe an accumulation close to high DMOS scores, which correspond to images with low perceived quality.

Figure 35: Reference images in the MULTI database.

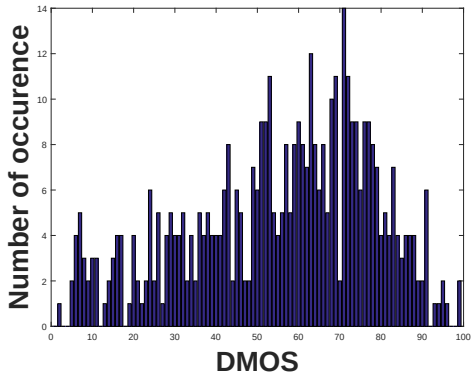


Figure 36: Normalized histograms of subjective scores in the MULTI database.

4.1.3 The TID 2013 Database

To increase the variety of artifacts, we use the TID 2013 (TID13) database [93]. Reference images are shown in Fig. 37, which is composed of 24 natural scene images and an artificially created image. Natural scene images have different content and textural characteristics. Each image was degraded with 24 types of distortion that have 5 distortion levels. Therefore, there are 3000 ($25 \times 24 \times 5$) images in the TID13 database. The resolution of the images are 512×384. In the subjective tests, a tristimulus approach was followed, in which three images were displayed and the participants were asked to select the better quality image out of the two distorted images. In a single experiment, distorted images corresponding to one of the reference

Figure 37: Reference images in the TID13 database.

images were displayed. Because of 24 distortion types and 5 distortion levels, there are 120 images, all of which participated in nine pair-wise comparisons. In each comparison, one point was assigned to the preferred image. Points for each comparison were summed up to obtain a quality score. Since there are 120 distorted images, 9 pair-wise comparisons, and 2 displayed images at the same time ($120 \times 9/2$), 540 pair-wise comparisons were performed corresponding to a single reference image by each participant. The distribution of the MOS scores are shown in Fig. 38. The majority of the scores are distributed between 0 and 7. There are only few instances, in which MOS scores are higher than 7. There is an accumulation close to high MOS scores, which corresponds to images with high perceived quality as in the LIVE database. Subjective experiments were conducted using various displays including LCD and CRT monitors, which were mostly 19 inch with a resolution of 1152×864 pixels. Users were asked to locate themselves based on their comfort level, which does not satisfy ITU-T recommendations. However, the authors in [93] claim that a constant

location would be against a real life scenario whereas an adaptive location would be more realistic.

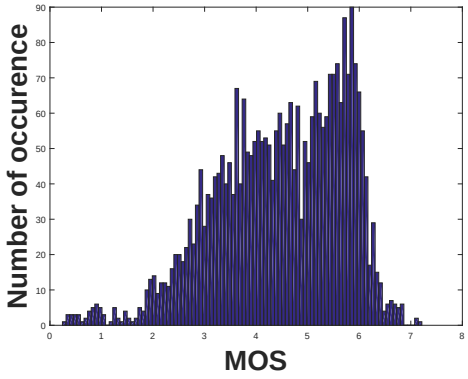


Figure 38: Normalized histograms of subjective scores in the TID13 database.

Distortion types, source of these distortions in practice, and visual system characteristics that are affected by these distortion types are summarized in Table 8. Images were degraded with additive zero-mean noise, which was modeled as white Gaussian noise. Noise was also non-uniformly added to chroma channels in the YCbCr color domain to test the difference between perception of noise in brightness and color channels. Images can be corrupted by spatially correlated noise because of processes including demosaicing and interpolation. Therefore, low-pass spatially correlated noise was also included in the database. Masked noise and high frequency noise were also used to simulate degradations based on image compression and digital watermarking. Local contrast sensitivity and spatial frequency sensitivity of a human visual system should affect the perception of these noise types. Coding and decoding errors in data transmission were simulated by uniformly distributed impulse noise [110]. Image registration and gamma correction artifacts were modeled by quantization noise and Gaussian blur was also used for image registration artifacts. Images were degraded with independent and identically distributed Gaussian noise and denoised with a method based on the 3D DCT [111] to obtain the images in image denoising category. Compression artifacts were generated with JPEG

Table 8: Distortion types, correspondence to practical situations, and characteristics affected by the human visual system in the TID13 database.

Distortion Type	Practical Correspondence	HVS Characteristics
Additive Gaussian noise	Acquisition	Adaptivity, robustness
Additive noise (intensive in color)	Acquisition	Color sens.
Spatially correlated noise	Digital photography	Spatial freq. sens.
Masked noise	Compression, watermarking	Local cont. sens.
High freq. noise	Compression, watermarking	Spatial freq. sens.
Impulse noise	Acquisition	Robustness
Quantization noise	Registration, gamma correction	Color, local cont., spatial freq.
Gaussian blur	Registration	Spatial freq. sens.
Image denoising	Denoising	Spatial freq., local cont.
JPEG compression	JPEG compression	Spatial freq. sens.
JPEG2000 compression	JPEG2000 compression	Spatial freq. sens.
JPEG transmission errors	Data transmission	Eccentricity
JPEG2000 transmission errors	Data transmission	Eccentricity
Non eccentricity pattern noise	Compression, watermarking	Eccentricity
Local block-wise distortions	Inpainting, acquisition	Evenness of distortions
Mean shift (intensity shift)	Acquisition	Light level sens.
Cont. change	Acquisition, gamma correction	Light level, local cont. sens.
Change of color saturation	Compression, Acquisition	Color sens.
Multiplicative Gaussian noise	Acquisition, Denoising	Adaptivity, robustness
Comfort noise	Image compression	Eccentricity
Lossy compression of noisy images	Compression, Denoising	Spatial freq. and local cont. sens.
Image color quantization with dither	Registration	Color sens., local cont., spatial freq.
Chromatic aberrations	Acquisition	Color sens. local cont. sens.
Sparse sampling and reconstruction	Compression, reconstruction	Spatial freq. and local cont. sens.

and JPEG20000. Moreover, transmission channel-based errors were introduced in the decoding of these compressed representations to obtain transmission artifacts. Non eccentricity pattern noise was simulated by randomly selecting patches of size 15×15 pixels and copying these patches to spatially close locations. Local block-wise distortions were modeled by randomly placing 32×32 patches of a random color over original images. Distortion levels of local block-wise degradations were adjusted by the number of patches and the similarity of the patch color with respect to the mean

color of the original patch. Mean shift and contrast change were also included in the image quality database. The distortion types described so far were introduced in the TID 2008 database [112].

In addition to the distortion types in the TID 2008 database, seven new categories were included in the TID 2013 database [93], which can be sorted as: change of color saturation, multiplicative Gaussian noise, comfort noise, lossy compression of noisy images, image color quantization with dither, chromatic aberrations, and sparse sampling and reconstruction. Change of color saturation corresponds to artifacts based on image acquisition and processing. Specifically, coarse quantization of color channels through the JPEG compression and printing issues can lead to these artifacts. To simulate color saturation artifacts, RGB images were converted to YCbCr images, and the values of the chroma channels were altered with a weighted linear sum to change the colorfulness of images. Multiplicative Gaussian noise was applied to each color channel in the RGB domain independently with the same standard deviation. To simulate comfort noise, RGB images were transformed into YCbCr images, each channel was lossy compressed with ADCTC [113], decompressed, and post-processed for blocking artifact removal. Reconstructed channels were subtracted from the original versions of the channels and these residual channels were subtracted form the reconstructed channels. Finally, these channels were combined and mapped from the YCbCr color domain to the RGB color domain. Lossy compression of noisy images was simulated by applying additive white Gaussian noise to each color channel and performing a lossy compression [113], whose quantization step was set to 1.73 times the standard deviation of the additive noise. Image color quantization with dither, which can be observed in image printing, was simulated by converting RGB images into indexed images using dither. Chromatic aberrations, which can be caused by image acquisition or transformation artifacts, were simulated by mutual shifting of

R, G, and B channels and bluring shifted channels. Compresive sensing-based artifacts were simulated by the method in [114], which was applied to each channel in the YCbCr domain. At first, a $2D\ DCT$ is performed over each color channel and some of the coefficients were assigned zero values to degrade an image. Inverse DCT was performed and BM3D filter [115] was applied over each channel. Filtered image was transformed to the DCT domain and compared with the initial DCT representation to restore non-zeroed DCT coefficients. This procedure was repeated for ten iterations and the final DCT representation was transformed back to the spatial domain to obtain a reconstructed image.

4.1.4 Analysis of the Databases

There are five distortion categories in the LIVE database and each category corresponds to a different distortion type. In MULTI, there are two distortion categories and each category includes two degradation types applied simultaneously. In TID13, there are six different distortion categories and each category includes various distortion types, whose number ranges from three to eleven. To analyze the performance of image quality estimation with respect to distortion types over different databases, we reclassify the distortion types in the databases LIVE, MULTI, and TID13 into seven categories. The `Compression` category includes JPEG, JP2K, and lossy compression of noisy images. The `Noise` category contains white noise, adaptive Gaussian noise, additive noise in chroma channels, impulse noise, spatially correlated noise, masked noise, high frequency noise, quantization noise, image denoising, multiplicative Gaussian noise, comfort noise, and lossy compression of noisy images. The `Communication` category includes Rayleigh fast-fading channel model, JPEG and JPEG2000 transmission errors. The `Blur` category includes Gaussian blur and sparse sampling and reconstruction error. The `Color` category contains color saturation change and color quantization with dither and chromatic aberrations. The `Global` category includes

intensity shift and contrast change. The `Local` category contains non-eccentricity pattern and local block-wise distortion of different intensity. The number of images in each category is summarized in Table 9.

Table 9: The number of distorted images per degradation type in each database.

	LIVE [106]	**MULTI [107]**	**TID13 [93]**	**Total**
Compression	460	180	375	1015
Noise	174	180	1375	1729
Communication	174	-	250	424
Blur	174	315	250	739
Color	-	-	375	375
Global	-	-	250	250
Local	-	-	250	250

4.2 Performance Metrics and Auxiliary Formulation

In this section, we briefly describe the performance metrics used in evaluating perceived image quality estimation performance.

4.2.1 Linearity, Ranking, Accuracy, and Consistency

4.2.1.1 Linearity

The Pearson correlation coefficient is used to measure the linearity of the predictions which is formulated as

$$PLCC = \frac{\sum_{s=1}^{T}(x_s - \mu_x)(y_s - \mu_y)}{\sqrt{\sum_{s=1}^{T}(x_s - \mu_x)^2} \cdot \sqrt{\sum_{s=1}^{T}(y_s - \mu_y)^2}}, \tag{30}$$

where x_s is the estimated score and y_s is the mean opinion score corresponding to an image indexed with s, μ is the average operator, and T is the total number of images.

4.2.1.2 Ranking

The Spearman correlation coefficient is used to measure the monotonic relationship between quality estimates and subjective scores. Instead of using exact values, ranking of the values is used. For example, let's assume that we have T images with

corresponding mean opinion scores (y_s). Based on the rankings, the minimum score should be ranked as 1, the maximum as T, and the others should lie in between 1 and T based on their rankings. This procedure is applied to both subjective scores and estimates. If the relative order of the subjective scores and the objective estimates are the same, then the correlation should be 1.0 otherwise it should be lower. The formulation of the Spearman correlation coefficient is given as

$$SRCC = 1 - \frac{6\sum_{s=1}^{T}(X_s - Y_s)^2}{T \cdot (T^2 - 1)}, \tag{31}$$

where X_s is the rank assigned to the score x_s and Y_s is the rank assigned to the subjective score y_s, which corresponds to an image indexed with s, and T is the total number of images.

The Kendall rank correlation coefficient is also based on ranking but we do not directly assign rankings to all estimates and scores. Instead, estimates and scores are compared one by one. For example, x_s is the estimate and y_s is the mean subjective score corresponding to an image indexed with s, and we have x_l and y_l corresponding to an image indexed with l. If $x_s > x_l$ and $y_s > y_l$ or $x_s < x_l$ and $y_s < y_l$, these pairs are denoted as `concordant`. If $x_s > x_l$ and $y_s < y_l$ or $x_s < x_l$ and $y_s > y_l$, these pairs are denoted as `discordant`. Finally, if $x_s = x_l$ and $y_s = y_l$, this pair is neither concordant nor disconcordant. Once all of the pair combinations are considered, the Kendall correlation coefficient is calculated as

$$KRCC = \frac{(T_{cor}) - (T_{dis})}{0.5 \cdot T \cdot (T - 1))}, \tag{32}$$

where T_{cor} is the number of concordant pairs, T_{dis} is the number of discordant pairs, and T is the number of images in a set.

4.2.1.3 Accuracy

The root mean squared error (RMSE) is used as the absolute prediction error to quantify accuracy. We take the difference between an individual subjective score

and an estimated quality score for each of the visual stimuli, take the square of this difference, calculate the mean squared error, and take the square root of the mean. The formulation of root mean squared error is

$$RMSE = \sqrt{\frac{\sum_{i=1}^{T}(x_s - y_s)^2}{N}}, \tag{33}$$

where x_s is the estimated score and y_s is the mean opinion score that correspond to an image indexed with s, and T is the total number of images in a test set.

4.2.1.4 Consistency

The outlier ratio is used as a consistency measure, which is formulated as

$$or = \frac{T_{out}}{T}, \tag{34}$$

where T is the total number of images and T_{out} corresponds to the total number of outliers. An outlier is defined as a point whose error exceeds the confidence interval of a mean opinion score. In the outlier ratio calculation (34), we count the number of outliers that are more than two standard deviations away from the average subjective score.

4.2.2 Regression, Statistical Significance, and Histogram Differences

4.2.2.1 Regression

In the validation of image quality assessment, monotonic regression is commonly used before measuring linearity, accuracy or consistency. We use the regression formulation in [116], which is formulated as

$$V = \beta_1 \left(\frac{1}{1} - \frac{1}{2 + \exp(\beta_2(V_0 - \beta_3))} \right) + \beta_4 V_0 + \beta_5, \tag{35}$$

where V_0 is the objective score, V is the regressed score, and the βs are parameters that are tuned based on the relationship between quality estimates and mean opinion scores.

4.2.2.2 Statistical Significance

We use statistical tests to evaluate the significance of performance differences in terms of correlation. To use the statistical significance tests, we assume that distributions of the visual quality scores follow a normal distribution. There are two main hypothesis in a statistical significance test. The first one (H_0) claims that there is no significant difference between correlation coefficients and the second one (H_1) claims that there is a significant difference between correlation coefficients.

In order to verify whether H_0 is true or not, at first, we assume that H_0 is true. Then, we perform required computations to check whether results contradict with the requirements of the hypothesis or not. We directly follow the steps described in ITU-T Rec. P.1401 [117]. At first, we calculate the Fisher z-transforms of compared correlation coefficients. The Fisher-z transform formulation is given as

$$z = 0.5 \cdot \ln\left(\frac{1+R}{1-R}\right), \tag{36}$$

where R is the correlation value and ln is the natural logarithm operation. When we calculate the Fisher z-transforms, we obtain z_1 and z_2 that correspond to compared correlation coefficients. Since we assume that H_0 hypothesis is true, the mean value of the difference between z_1 and z_2 would be zero as

$$\mu_(z_1 - z_2) = 0. \tag{37}$$

The standard deviation of the difference formulation is given as

$$\sigma_(z_1 - z_2) = \sqrt{\sigma_{z_1}^2 + \sigma_{z_2}^2}, \tag{38}$$

where σ_{z_1} and σ_{z_2} are the standard deviations for compared correlation coefficients. The standard deviation of the Fisher-z statistic is given as

$$\sigma_z = \sqrt{\frac{1}{T-3}}, \tag{39}$$

where T is the total number of images used in the calculation of correlation coefficients. Since we know the Fisher z-transform values, the mean, and the standard deviation of the difference, we can calculate the significance value (Z_t) as

$$Z_t = \frac{z_1 - z_2 - \mu_(z_1 - z_2)}{\sigma(z_1 - z_2)}. \tag{40}$$

If Z_t is below the two-tailed t value, H_0 is true. In case Z_t is larger, H_1 is true. The t-value is a function of the degrees of freedom of the distribution and the confidence interval. Previously, printed lookup tables were used to find the t-test values. However, recently, software packages are commonly used. There are various software packages that are used for statistical computing but the most common ones are R ® and the toolboxes in Matlab ®.

4.2.2.3 Histogram Differences

To measure the difference between subjective scores and objective scores, we measure the difference between normalized histograms of subjective scores and regressed quality estimates through common histogram difference metrics including Earth Mover's Distance (EMD), Kullback-Leibler (KL) divergence, Jensen-Shannon (JS) divergence, histogram intersection (HI), and l_2 norm.

CHAPTER V

NOVEL IMAGE QUALITY ESTIMATORS

Visual representations capture perceived degradations in different manners. Retinal ganglion cell- and cortical neuron-based representations concentrate more on structural degradations whereas color differences and color name distances focus more on chromatic degradations as discussed in Chapter 3. Therefore, we need to fuse various visual representations to obtain a universal quality estimator. Based on these visual representations, we introduce two novel image quality estimators `PerSIM` [1] and `CSV` [2], and a new image quality-assistance method `BLeSS` [3] as described in Sections 5.1, 5.2, and 5.3. We combine our findings from visual system characteristics and color perception with data-driven approaches to directly obtain visual representations and measure their contribution to perceived quality. The majority of existing data-driven methods require subjective scores or degraded images in the training. In contrast, we follow an unsupervised approach trained with generic images. We introduce a novel unsupervised image quality estimator `UNIQUE` [4]. Moreover, we extend `UNIQUE` with multiple models and layers to obtain `MS-UNIQUE` [5] and `DMS-UNIQUE` as described in Section 5.4. In Sections 5.5 and 5.6, we analyze the performance of introduced image quality-assistance method `BLeSS`, and quality assistance methods `PerSIM`, `CSV`, `UNIQUE`, `MS-UNIQUE`, and `DMS-UNIQUE`. We show that introduced quality estimators consistently lead to lower error and outlier ratio and higher Pearson and Spearman correlation compared to majority of the existing methods tested in the literature, and quality-assistance leads to significant performance enhancement in estimating the perceived quality under color-based degradations.

5.1 PerSIM: Perceptual Similarity

The introduced method `PerSIM` is a full-reference image quality estimator, whose block diagram is given in Fig. 39. Initially, a reference and a distorted image are in the RGB color domain. First, these images are transformed to the La*b* color space. Chroma channels are fed into pixel-wise similarity blocks and lightness channels are fed into Laplacian of Gaussian (`LoG`) blocks that compute the pixel-wise similarity of `LoG` maps. The same operations are repeated for different resolutions. The feature maps obtained from different resolutions are interpolated to the original image size and combined using geometric mean operations. Finally, we perform a sensitivity adjustment and a pooling operation to obtain the quality score `PerSIM`.

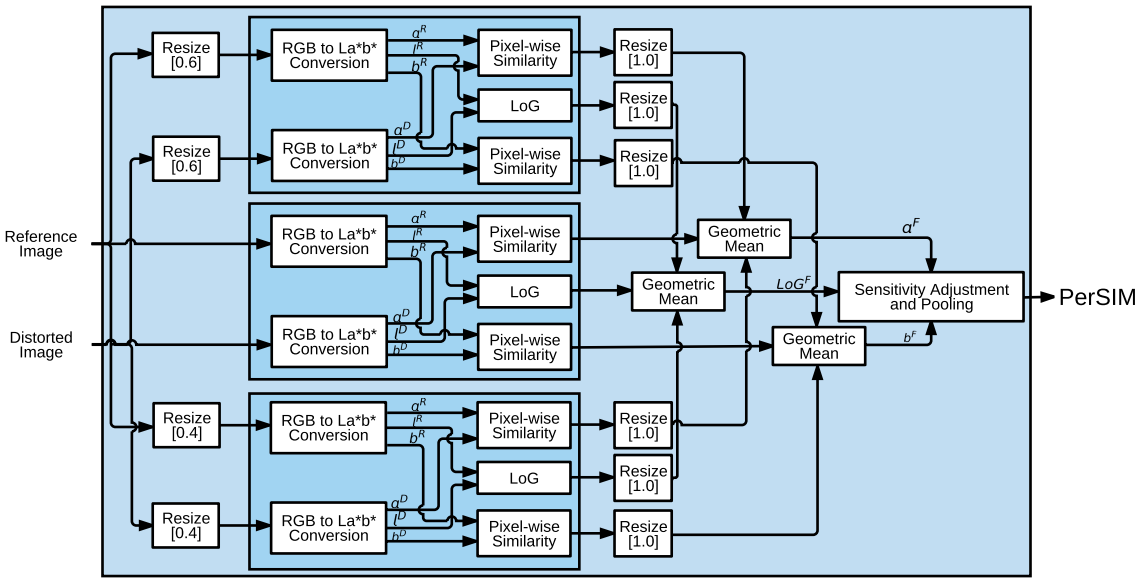


Figure 39: PerSIM block diagram.

5.1.1 Main Blocks

To decorrelate color- and structure-based information, we transform RGB images into the La*b* color domain. Pixel-wise fidelity is used to measure color-based degradations in the chroma channels a* and b*. Because color perception is only a part of the overall perception, we also measure the degradations in the lightness channel,

which contains structural information. In order to obtain a perceptual quality estimator, we need to introduce visual system models. In `PerSIM` [1], we use a Laplacian of Gaussian (LoG) operator to partially model the contrast sensitivity mechanism in retinal ganglion cells. Lightness channels of compared images are filtered using LoG operators to obtain LoG feature maps. To obtain the similarity between these feature maps, we use a similarity index that satisfies symmetry, boundedness, and unique maximum characteristics. We formulate the similarity index as

$$LoG_i = \frac{2 \cdot LoG_i^R \cdot LoG_i^D + c_1}{(LoG_i^R)^2 + (LoG_i^D)^2 + c_1}, \tag{41}$$

where i is the pixel index, LoG^R is the LoG map of the reference image, LoG^D is the LoG map of the distorted image, LoG_i is the similarity index corresponding to pixel index i, c_1 is a constant, and channel index is not explicitly shown because LoG maps are computed solely over lightness channels. The numerator in the similarity expression corresponds to projecting one representation onto the compared representation, and the denominator can be considered as a normalization term. We use a similarity expression instead of a difference formulation because we want to quantify relative changes that are perceived rather than absolute changes as in Weber's law [118].

Chroma similarity in the a* and the b* channels are computed as in Eq. (41) by replacing LoG feature maps with chroma maps. These chroma-based similarity maps are denoted as a and b. Since perception in a visual system is hierarchical and visual representations can have different abstraction levels, we calculate the similarity maps at different resolutions and fuse them together as follows:

$$LoG_i^F = \sqrt[3]{LoG_i^{1.0} \cdot LoG_i^{0.6} \cdot LoG_i^{0.4}}, \tag{42}$$

$$a_i^F = \sqrt[3]{a_i^{1.0} \cdot a_i^{0.6} \cdot a_i^{0.4}}, \tag{43}$$

$$b_i^F = \sqrt[3]{b_i^{1.0} \cdot b_i^{0.6} \cdot b_i^{0.4}}, \tag{44}$$

where subscript i is the pixel index, the superscripts 0.4, 0.6, and 1.0 correspond to the ratio between the resolution of that similarity map with respect to the resolution of the input images, F superscript corresponds to the similarity maps obtained from fusing resized similarity maps, and $\cdot$ is the pixel-wise multiplication operator. We use a geometric mean operation instead of an arithmetic operation to combine similarity maps at different resolutions. Because the resizing operation can slightly change the numerical ranges of similarity maps and an arithmetic mean would be a biased estimator towards the similarity maps with higher numerical ranges.

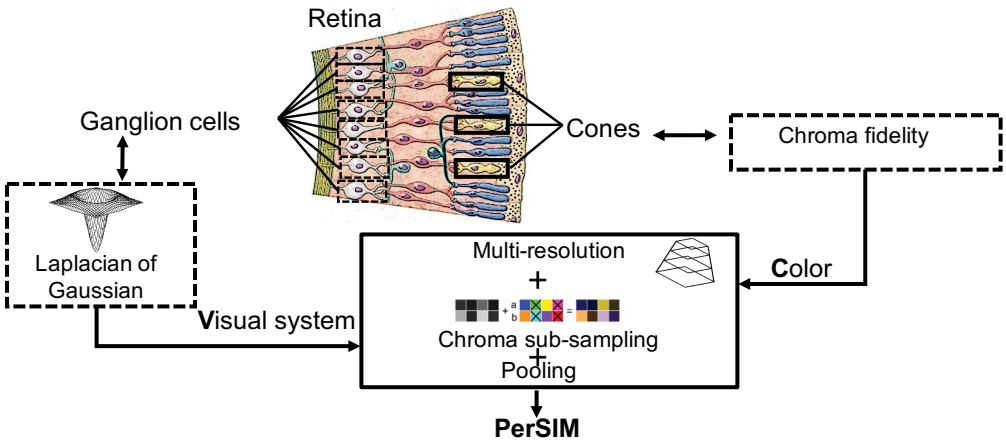


Figure 40: Graphical abstract of PerSIM.

In image and video coding, chroma subsampling is introduced to assign fewer bits per pixels to chroma channels. 4:2:2 is one of the most commonly used subsampling format, in which a chroma channel is assigned half the bit budget of a luma channel [27]. In the introduced work, we use the relative significance of channels in 4:2:2, and tune the power of LoG similarity to 4.0 and the powers of similarities for chroma channels to 2.0. After this sensitivity adjustment, we obtain the pixel-wise minimum among the similarity indices as follows

$$Lab_i^F = min((LoG_i^F)^4, (a_i^F)^2, (b_i^F)^2), \tag{45}$$

where the intuition is based on the fact that perceived quality is dominated by the

most significant degradation, which corresponds to the lowest quality value. Finally, a mean pooling operation is performed over sensitivity-adjusted perceptual quality map as

$$PerSIM = \left(\sum_{i=1}^{M \cdot N} \frac{Lab_i^F}{M \cdot N} \right)^{c_2}, \tag{46}$$

where i is the window index corresponding to the upper left corner pixel index, M is the number of columns and N is the number of rows in an input image, and c_2 is a constant to set the range of quality estimates. A graphical abstract of PerSIM is given in Fig. 40, which highlights retina, Laplacian of Gaussian (LoG) model of ganglion cells, multi-resolution, and color characteristics.

5.2 CSV: Color, Structure, and Visual System

The second method we introduce is called `CSV`, which is a full-reference image quality estimator, whose block diagram is given in Fig. 41. Initially, the reference (I^R) and the distorted (I^D) images are represented in the RGB color domain. First, color channels of these RGB images are separated and fed into Laplacian of Gaussian (`LoG`) and normalization blocks in parallel. For each channel, the output of the `LoG` blocks are fed into absolute difference blocks to obtain retinal ganglion cell-based difference (`RGCD`) maps. The geometric mean of these maps is computed to obtain a final `RGCD` map. Similarly, separated RGB channels are fed into the normalization and the absolute difference blocks to obtain structural difference (`SD`) maps, which are combined with a geometric pooling operation to obtain a final `SD` map.

To obtain color difference and color name distance maps, we perform a mean pooling operation over the reference and the distorted images. These mean pooled images are transformed to the LCH and the La*b* color domains, in which chroma and luma channels are decorrelated. The mean pooled La*b* maps are fed into color name blocks to obtain color descriptors. The Earth Mover's Distance (`EMD`) between descriptors is calculated for each pixel to obtain a color name distance map (`CND`),

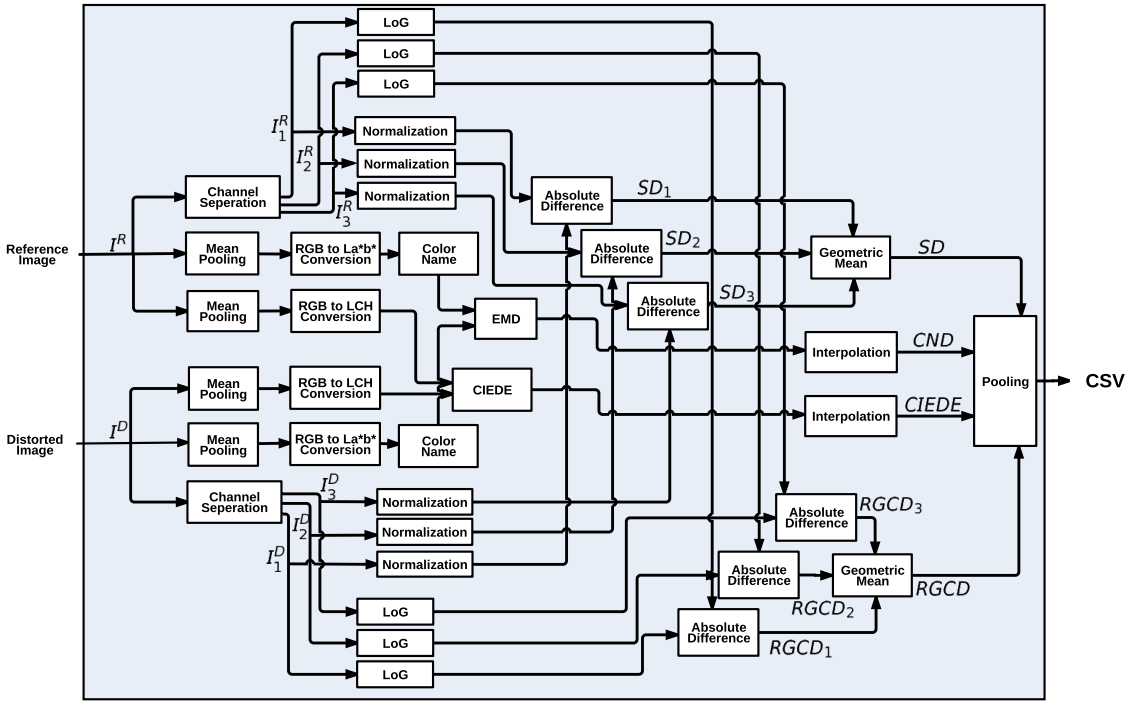


Figure 41: CSV block diagram.

which is interpolated to the same size of the original images. Mean pooled LCH maps are fed into `CIEDE` blocks to obtain a color difference map. This difference map is interpolated to the same size of the original images to obtain a `CIEDE` map. Finally, all feature maps are pooled together to obtain an estimated quality score, which is denoted as `CSV`. A graphical abstract of CSV is given in Fig. 42, which shows the interrelationships among the retina, Laplacian of Gaussian (LoG) model of ganglion cells, CIEDE 2000 formulation, color names, and normalization-based suppression model of cortical neurons.

5.2.1 Quality Map Fusion and Spatial Pooling

In order to measure the perceived differences in colors, neither fidelity-based chroma similarity nor color difference equations are sufficient. Therefore, we use color name distances to complement color difference equations. Since color perception is not sufficient to model general perception, we use partial models of contrast sensitivity in

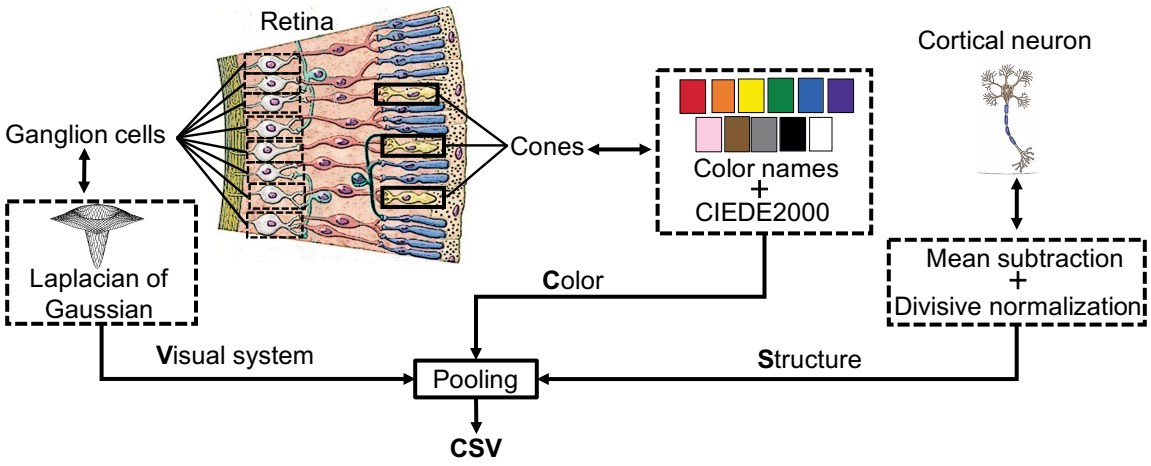


Figure 42: Graphical abstract of CSV.

ganglion cells and suppression mechanisms in cortical neurons to obtain the quality estimator `CSV` [2].

RGCD and SD quality estimator blocks are calculated over every color channel to detect visual degradations. Because the range of these blocks over different color channels depends on the color distribution, arithmetic average over color channels would be a biased estimator towards highly populated color channels. Therefore, we calculate the geometric mean over color channels to obtain final difference maps as

$$SD_i = \sqrt[3]{SD_{i,1} \cdot SD_{i,2} \cdot SD_{i,3}}, and \tag{47}$$

$$RGCD_i = \sqrt[3]{RGCD_{i,1} \cdot RGCD_{i,2} \cdot RGCD_{i,3}}, \tag{48}$$

where `RGCD` is the retinal ganglion cell-based difference map and `SD` is the structural difference map.

Color-based (`CIEDE`, `CND`) and visual system-based (`RGCD`, `SD`) maps need to be pooled to obtain a final quality estimate. If the maps were normalized to the same range, we would be able to use additive fusion to obtain an estimate. However, calculating the statistics on tested databases and normalizing estimates according to this side information would not be a fair approach since the evaluation should not

have any information other than a reference and a distorted image. Therefore, we perform a multiplicative fusion operation to combine the feature maps of individual blocks with different scalar ranges as follows

$$CSV_i = RGCD_i \cdot SD_i \cdot (A \cdot CND_i + (1 - A) \cdot CIEDE_i), \tag{49}$$

where $\cdot$ is the pixel-wise multiplication operator and A is the weight that adjusts the ratio of color difference and color name distance. All the quality estimator blocks that we use are based on difference or distance operators, which lead to high values in case of significant degradations. However, we want to propose a quality estimator that leads to high values in case of high quality and low values under significant degradations. Thus, the residual of the average distortion can be calculated. In order to adjust the scalar range of the quality estimator, the residual of the n^{th} root is calculated to obtain the final quality score as

$$CSV = 1 - \sqrt[c_3]{\frac{1}{M \cdot N} \sum_{i=1}^{M \cdot N} CSV_i}, \tag{50}$$

where i is the window index, M is the number of columns, N is the number of rows in an input image, and c_3 is the power of the root operator, which is generally referred as n. Structural difference (SD) and retinal ganglion cell-based difference (RGCD) maps are calculated over full resolution images whereas CIEDE and color name distance (CND) maps are calculated over low resolution images and resized to the original resolution using a bicubic interpolation operation. Therefore, when a `CSV` vector is transformed into a 2-D representation, we directly obtain a quality map.

5.2.2 Parameter Tuning and Complexity Analysis

5.2.2.1 Parameter Tuning

The parameters of `CSV` are summarized in Table 10. There are nine parameters, five of which (K_L, K_C, K_H, T, and N) are directly obtained from the original implementations of the used formulations. Two of the parameters (W and σ) are selected from

the visual assessment of quality maps. The parameter c_3 is selected based on the distribution of the quality estimates and the selection of the parameter A is based on the color chart toy example.

The `CIEDE` block has five parameters. W is the size of the rectangular window set to 20×20. K_L, K_C, and K_H are the correction parameters that are calibrated based on the experimental environment. These parameters are set to 1.0 in the CIEDE2000 color difference equation under the CIE standard observation conditions. The subjective test setup cannot exactly match the standard conditions and these parameters need to be tuned based on the environment. Meanwhile, the proposed method should work for any image, independent of acquisition or display technology. Therefore, we fix these environmental parameters to standard values. We use a threshold (T) to limit the CIEDE2000 estimates to small degradations and set T to 20 as in [119]. The window size (W) in `CIEDE`, `CND`, `SD`, and `RGCD` are set to 20×20 by visually assessing the distinctiveness of randomly selected feature maps. Smaller window size leads to amplification of noise and fluctuations whereas larger window results in the loss of local information, both of which decrease the prediction accuracy. The standard deviation (σ) is set to 50 in the retinal ganglion-cell based difference by visually assessing the feature maps.

The researchers in [97] investigate the basic color terms in verbal usage and 20 different languages are used to obtain universal categories independent of the language characteristics. As a consequence of these studies, the researchers defined 11 basic color groups, and the value of the parameter N is set to the number of basic color groups. The difference between the color name descriptors are also computed with variations of L-norms and information theoretic formulations. However, the performance of the quality estimator does not change significantly compared to Earth Mover's Distance. `CND` and `CIEDE` have different numerical ranges as discussed in

Table 10: Parameters in the CSV formulation.

CSV Block	Parameter	Description	Value
CIEDE	W	Window Size	$[20, 20]$
	K_L	CIE correction parameter	1.0
	K_C	CIE correction parameter	1.0
	K_H	CIE correction parameter	1.0
	T	Color degradation level threshold	20
CND	W	Window size	$[20, 20]$
	N	Number of color names	11
SD	W	Window size	$[20, 20]$
RGCD	W	Window size	$[20, 20]$
	σ	Standard deviation	50
Pooling	A	Linear combination weight	0.9
	c_3	Power value of the monotonic mapping function	4

Section 3.2. Therefore, we use weights to scale the color-based differences in the summation. The weight of CND is set to A and CIEDE to $(1 - A)$, where A is 0.9. We select the remaining parameter A by finding the weight in the color chart example that assigns a higher value to CND compared to CIEDE and results in higher differences for less similar colors. If we assign a lower weight to CND (0.8 instead of 0.9), weighted color difference formulation would not be able to detect the similar color tones in the second row.

A quality score is obtained using Eq. (50) by computing the root of an average CSV value, where c_3 is set to 4. The numerical range of CSV can be set to different values by using other monotonic functions or power values. However, scaling does not bias the performance of quality estimation since ranking-based performance metrics are not affected from a monotonic mapping, and regression would eliminate the effect of the mapping in terms of linearity. Therefore, parameter selection process is independent from the performance validation stage and this independence should

eliminate overfitting to the tested databases.

5.2.2.2 Complexity Analysis

We classify the main blocks in the introduced method according to their computational complexity. Channel separation, mean pooling, local normalization, absolute difference, geometric mean, and pooling operations are computationally less demanding compared to color space transformation, color name extraction, interpolation, EMD, CIEDE2000, and LoG filtering. We perform mean pooling over compared images with a 20×20 non-overlapping window to decrease the number of processed pixels by 400 times, and we perform most of the computationally intensive operations after size reduction. Moreover, we use a robust version of EMD [119] that is shown to be faster than the original version [99] by up to $75-700$ times in various applications. We can further reduce the computational time of `CSV` by modifying the interpolation method, the filtering operation, and the data processing mechanisms. The interpolation method does not significantly affect the performance of the introduced method. Therefore, a bilinear- or a nearest-neighbor-based interpolation can be used instead of a bicubic interpolation to reduce the overall computational complexity. Laplacian of Gaussian can be approximated with a difference of Gaussian operator, which can reduce the computational complexity. In the current implementation, `CND` and `CIEDE` values are computed for each pixel sequentially. These sequential processes can be parallelized to reduce the computation time.

5.3 BLeSS: Bio-inspired Low-level Spatiochromatic Similarity

We extend the saliency by induction model as a similarity method to assist full-reference image quality estimators that originally oversimplify color perception processes. As a proof of concept, the introduced assistant similarity method is used to

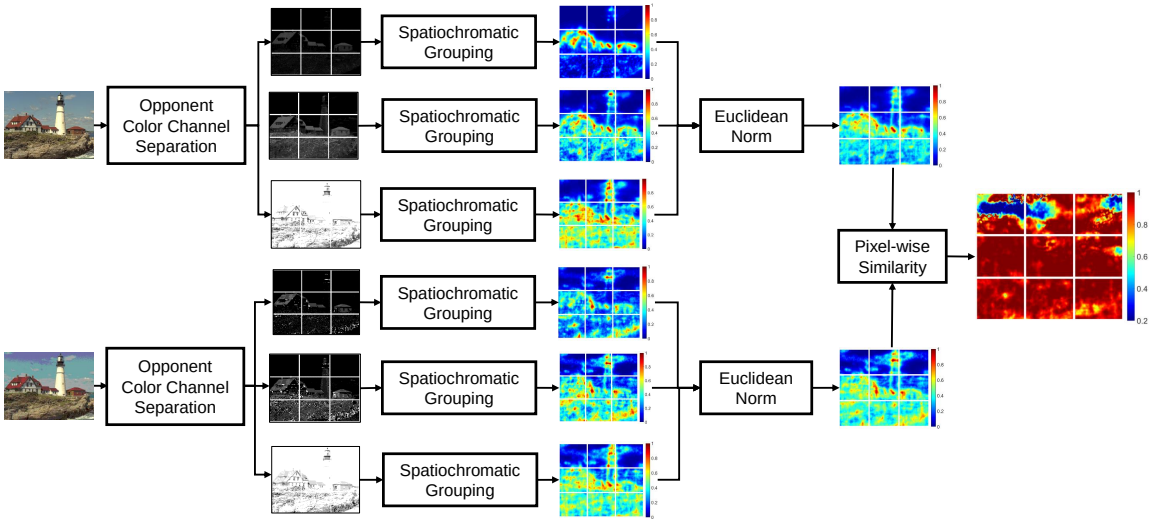


Figure 43: BLeSS block diagram with visualized feature maps.

complement image-quality estimators based on phase congruency, gradient magnitude, and spectral residual.

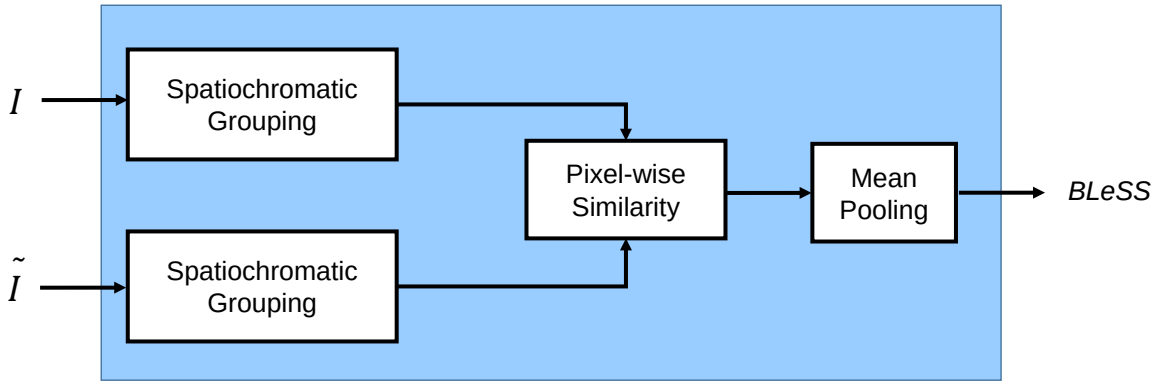


Figure 44: BLeSS block diagram.

We process reference and distorted images using spatiochromatic grouping pipeline and processed maps are fed to a pixel-wise similarity block as illustrated in Figs. 43 and 44. Then, we perform a mean pooling operation over the similarity map to obtain the bio-inspired low-level spatiochromatic similarity (`BLeSS`) score as shown in Fig.

44.

The similarity between feature maps is calculated with the familiar expression that has been used in most of the pixel-wise and structural similarity methods as

$$BLeSS = \frac{2 \cdot SG^R \cdot SG^D + c_4}{(SG^R)^2 + (SG^D)^2 + c_4}, \tag{51}$$

where SG^R is the spatiochromatic grouping map of a reference image, SG^D is the spatiochromatic grouping map of a distorted image, and c_4 is a constant added to denominator to avoid the issues when denominator converges to 0.0 and also to numerator to avoid a bias. Similarity score is 1.0 when compared feature maps are same and it gets closer to 0.0 as the difference between compared maps increases.

`BLeSS` is a partial model of low-level spatiochromatic similarity, which can be used to explain perceptual effects including spatial-frequency, spatial-orientation, and surround-contrast effects. Therefore, we propose `BLeSS` as an assistance mechanism to complement image quality estimators that originally oversimplify the role of color in perception.

5.3.1 State of the Art Quality Estimators Overlooking Color Perception

`BLeSS` is used to enhance feature similarity- (FSIM) and spectral residual-based (SR-SIM) quality estimators. In both of these quality estimators, similarity maps are masked with weight maps, which are introduced as representations of reliability, saliency, or region of interest. The intuition behind using a weight map is to assign significance to pixels so that when a similarity map is pooled into a final quality score, significant pixels would be more influential. We formulate the weighting operation as

$$\frac{\sum_{i=1}^{M \cdot N} S_i \cdot W_i}{\sum_{i=1}^{M \cdot N} W_i}, \tag{52}$$

where i is the pixel index, S is the similarity map, and W is the weight map. In the following sections, we use the formulation in Eq. (52) to describe the quality estimators FSIM and SR-SIM.

5.3.1.1 FSIM: Feature Similarity

Gradient magnitude is used in image-quality metrics [11, 35] to quantify local contrast. The most commonly used operators to calculate gradient magnitude are Sobel, Prewitt, and Scharr gradients. Sharp changes in intensity are captured by gradient magnitude but the significance of these changes is not quantified. However, phase congruency can be used to quantify the perceptual significance of changes. Since gradient magnitude and phase congruency are complementary, the authors in [11] combine them to assess image quality with a method denoted as feature-similarity index (FSIM).

The phase congruency formulation given in [120] is used in the FSIM method. The phase congruency description starts from a 1D signal x. F_e^n and F_o^n are denoted as the even-symmetric and the odd-symmetric filters on scale n, which are used to obtain a quadrature pair. Log-Gabor filters are used as the quadratic pair of filters in FSIM. Response of each quadrature pair is expressed as

$$[x_{e,i}^n, x_{o,i}^n] = [x_i * F_e^n, x_i * F_o^n], \tag{53}$$

where i is the index of the entities in the signal, n is the scale, $*$ is the convolution operator, x_e^n is the even decomposition of the signal x at scale n, and x_o^n is the odd decomposition of the signal x at scale n. Local amplitude is given as

$$A_i^n = \sqrt{(x_{e,i}^n)^2 + (x_{o,i}^n)^2}. \tag{54}$$

The phase congruency of a reference 1-D signal is computed as

$$PC_i^R = \frac{\sqrt{\left(\sum_n x_{e,i}^{R,n}\right)^2 + \left(\sum_n x_{o,i}^{R,n}\right)^2}}{\epsilon + \sum_n A_i^{R,n}}, \tag{55}$$

where PC^R is the phase congruency vector of a reference signal and ϵ is a constant that avoids instabilities in the denominator. Phase congruency expression for 1-D is performed over different orientations to obtain a 2-D version. A Gaussian formulation

is used as the spreading function to extend log-Gabor filter to 2-D. The 2-D phase congruency formulation of a reference image is expressed as

$$PC_i^R = \frac{\sum_j \sqrt{\left(\sum_n x_{e,i,\theta_j}^{R,n}\right)^2 + \left(\sum_n x_{o,i,\theta_j}^{R,n}\right)^2}}{\epsilon + \sum_n \sum_j A_{i,\theta_j}^{R,n}}, \tag{56}$$

where $\theta_j = j\pi/J,\ j = 0, 1, ..., J-1$ is the orientation angle of the filter and J is the number of orientations. The similarity between phase congruency maps can be calculated as

$$PC = \frac{2 \cdot PC^R \cdot PC^D + c_4}{(PC^R)^2 + (PC^D)^2 + c_4}, \tag{57}$$

where PC is the similarity map based on phase congruency, PC^R is the phase congruency map of a reference image, PC^D is the phase congruency map of a distorted image, and c_4 is a constant added to denominator to avoid the issues when denominator converges to 0.0.

Gradient magnitude is computed with the Sobel, the Prewitt, and the Scharr operator in the feature similarity index and the Scharr operator is used in the final metric because of outperforming performance. The Scharr gradient magnitude of an image is defined as $GM = \sqrt{(GM_x * I)^2 + (GM_y * I)^2}$ and the 2-D gradient operators are given as

$$GM_x = \frac{1}{16}\begin{bmatrix} 3 & 0 & -3 \\ 10 & 0 & -10 \\ 3 & 0 & -3 \end{bmatrix}, GM_y = \frac{1}{16}\begin{bmatrix} 3 & 10 & 3 \\ 0 & 0 & 0 \\ -3 & -10 & -3 \end{bmatrix}, \tag{58}$$

where the horizontal gradient magnitude operator GM_x is the transpose of the vertical gradient magnitude operator GM_y. The similarity between gradient magnitude maps can be calculated as

$$GM = \frac{2 \cdot GM^R \cdot (GM^D) + c_4}{(GM^R)^2 + (GM^D)^2 + c_4}, \tag{59}$$

where GM is the similarity map based on gradient magnitude, GM^R is the gradient magnitude map of a reference image, GM^D is the gradient magnitude map of a

distorted image, and c_4 is a constant added to denominator to avoid the issues when denominator converges to 0.0.

FSIM is calculated using the expression in Eq. (52), in which similarity map is expressed as

$$S = PC \cdot GM, \tag{60}$$

and weight map is defined as

$$W = max(PC^R, PC^D). \tag{61}$$

5.3.1.2 SR-SIM: Spectral Residual Similarity

Perceptual significance can be detected using saliency-based approaches such as spectral residual [121], which is based on the sensitivity of a visual system to unexpected changes as part of the suppression mechanisms. To obtain spectral residual visual saliency, an image is transformed from the spatial domain to the frequency domain using the Fourier transform ($\mathscr{F}$). Lightness or luma (l) channel of an image is used to obtain the spectral residual as

$$\left|L^R\right| = \left|\left(\mathscr{F}\left[I^R\right]\right)\right|, \tag{62}$$

where l^R is the lightness channel of the reference image and $|\cdot|$ is the magnitude operator. The phase component is given as

$$\underline{\angle L} = \underline{\angle(\mathscr{F}\left[I\right])}, \tag{63}$$

where $\underline{\angle\cdot}$ is the phase operator. The spectrum of a signal is computed by the log of the magnitude ($\log\left|L^R\right|$) and the average spectrum is computed by convolving the spectrum with an averaging filter (g). The difference between the spectrum and the averaged spectrum results in the residual, which is formulated as

$$RE^R = \log\left|L^R\right| - g * \log\left|L^R\right|, \tag{64}$$

where RE is the residual map, $*$ is the convolution operator, and g is the averaging filter. The residual is combined with the phase of the input image, and then an inverse Fourier transform is performed. A reconstruction operation using the residual is formulated as

$$SR^R = h * \mathscr{F}^{-1}\left[RE^R \angle L\right], \tag{65}$$

where $\mathscr{F}^{-1}$ is the inverse Fourier transform operator, h is the Gaussian low-pass filter used to smooth out a reconstructed map, and $[\cdot \angle]$ is the representation of a signal in terms of its magnitude and its phase.

The similarity between spectral residual maps is calculated as

$$SR = \frac{2 \cdot SR^R \cdot SR^D + c_4}{(SR^R)^2 + (SR^D)^2 + c_4}, \tag{66}$$

where SR^R is the spectral residual map of a reference image, SR^D is the spectral residual map of a distorted image, and c_4 is a constant added to denominator to avoid the issues when denominator converges to 0.0. SR-SIM is calculated using the expression in Eq. (52), in which feature map is expressed as

$$S = SR \cdot (GM)^{c_5}, \tag{67}$$

where c_5 is a parameter to adjust the relative importance of GM with respect to SR and weight map is defined as

$$W = max(SR^R, SR^D), \tag{68}$$

where SR^R is the spectral residual map of a reference image and SR^D is the spectral residual map of a distorted image

5.3.2 BLeSS-assisted Image Quality Assessment

In this subsection, we formulate the `BLeSS` assisted methods using the following notation: GM is the gradient magnitude similarity map, PC is the phase congruecny similarity map, and SR is the spectral residual similarity map. GM^R is the gradient

Table 11: Parameters in the similarity formulation of BLeSS.

Similarity	Metric	Coefficient c_4
GM	FSIM	160
	SR-SIM	225
PC	FSIM	0.85
SR	SR-SIM	0.4
τ	BLeSS	0.4

magnitude map of a reference image, PC^R is the phase congruency map of a reference image, SR^R is the spectral residual map of a reference image, and feature maps corresponding to distorted images are GM^D, PC^D, and SR^D.

- `BLeSS`-assisted FSIM similarity map (BLeSS-FSIM) is defined as

$$S = GM \cdot PC \cdot BLeSS, \tag{69}$$

  where $\cdot$ is the pixel-wise multiplication operator. The BLeSS-FSIM weight map is given by

$$W = max(PC^R \cdot SG^R, PC^D \cdot SG^D). \tag{70}$$

- `BLeSS` assisted SR-SIM similarity map (BLeSS-SR-SIM) is defined as

$$S = SR \cdot (GM \cdot BLeSS)^{c_6}, \tag{71}$$

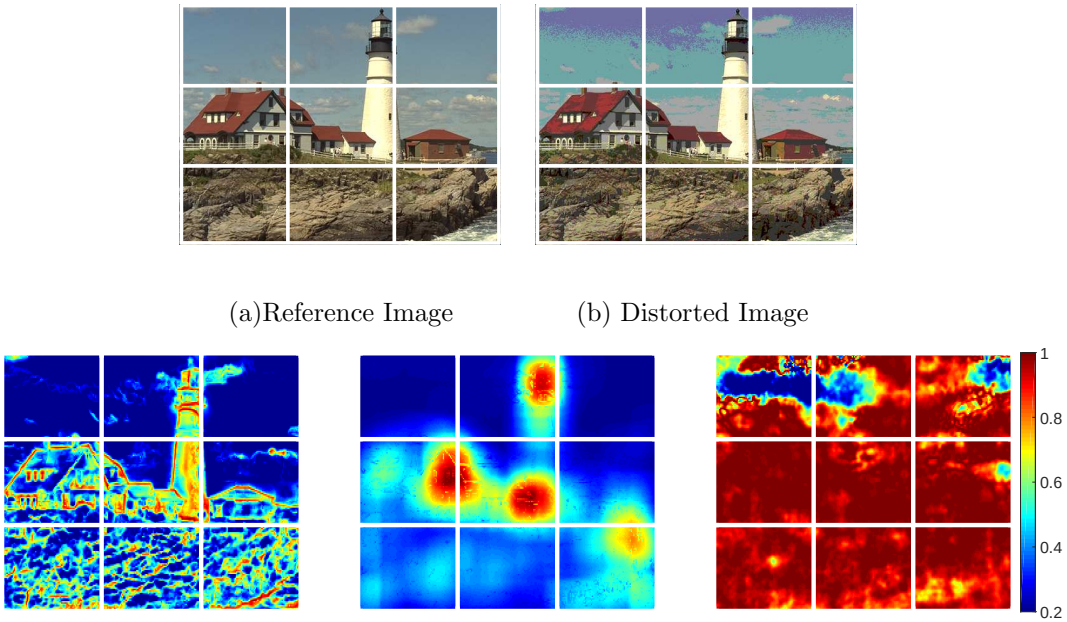
  where c_6 is a constant set to 0.5 in the original implementation [122]. The BLeSS-SR-SIM weight map is given by

$$W = max(SR^R \cdot SG^R, SR^D \cdot SG^D). \tag{72}$$

5.3.2.1 Parameter Setup

Similarity maps based on the gradient magnitude, the phase congruency, the spectral residual, and the low-level spatiochromatic grouping are computed by substituting the feature maps in Eq. (51). We use the original parameters in the publicly available

codes for FSIM [123] and SR-SIM [122]. The parameters in Eq. (51) for different feature maps are summarized in Table 11. The `BLeSS` parameter is set to the same value with the SR-SIM parameter without any tuning.



(a)Reference Image (b) Distorted Image

(c)FSIM Map (d) SR-SIM Map (e) BLeSS Map

Figure 45: Reference image, distorted image, and similarity maps of FSIM, SR-SIM, and BLeSS.

5.3.2.2 Visualization

The quality maps of FSIM, SR-SIM, and BLeSS corresponding to the images in Fig. 45 (a-b) are given in Fig. 45 (c-e). In this visualization, high values correspond to minor level degradations and low values indicate significant degradations. The color goes from red to blue under significant degradation as shown in the color bar. We show weighed similarity maps that correspond to the numerator of the expression in Eq. (52). All images are shown with a grid structure to make the visual comparison easier among quality maps. We normalize all similarity maps by subtracting the mean

and dividing by the maximum to visually highlight the difference between similarity maps.

Degradations based on color and structure are significant in the top row. Sharp tone changes and pixel-wise discontinuities in the sky are easily perceived as well. In the middle row, we can observe degradation over roofs of houses and around windows where we have edges or sharp transitions. However, it is not easy to observe degradations around regions with over exposure such as the wall of the lighthouse. Degradations are less perceivable around the highly textured regions as observed in the bottom grids where we have the textured rock components.

The `BLeSS` map captures the degradations in the sky region, especially around the tone change, whereas it overlooks the degradations in other regions. FSIM captures the degradations that are overlooked by `BLeSS` but FSIM also captures pixel-wise changes that are not even perceived because of the masking effect around highly textured regions. SR-SIM detects some of the degradations but it is not very sensitive to the level of degradations. SR-SIM identifies four regions as high quality and the rest as low quality in this visual example. We use `BLeSS` as a color-based assistance mechanism to FSIM and SR-SIM because `BLeSS` detects the color tone changes in the sky region, which are overlooked by FSIM and SR-SIM.

5.4 UNIQUE: Unsupervised Image Quality Estimation

In `PerSIM`, `CSV`, and `BLeSS`, we handcraft image quality estimators based on visual system characteristics and color information. In `UNIQUE`, we still utilize color information and visual system characteristics. However, instead of handcrafting, we follow a learning-based approach.

5.4.1 Preprocessing

Generic images are preprocessed to obtain effective and descriptive spatial representations. Preprocessing steps are illustrated in Fig. 46. At first, a color space selection

is performed. Patches are randomly sampled over selected color channels, concatenated into a single vector, and normalized using a mean subtraction and a whitening operation.

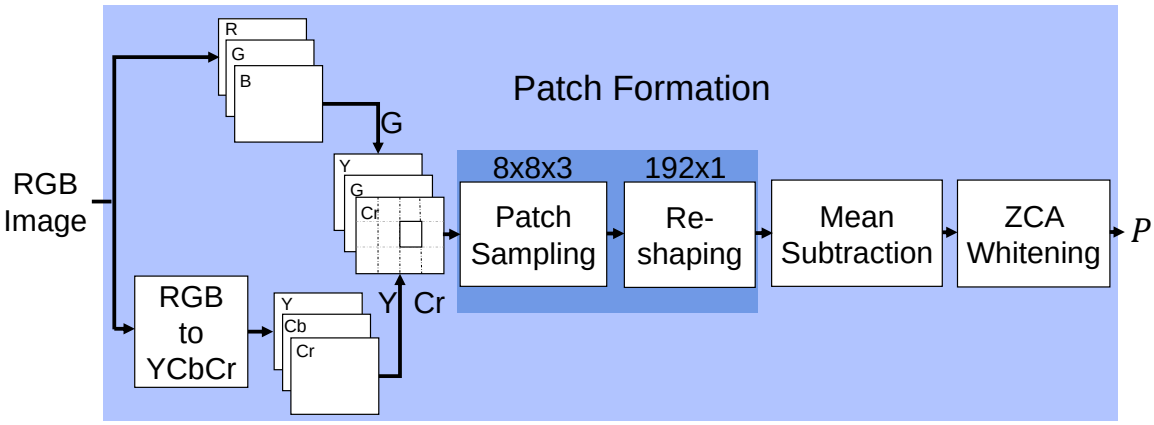


Figure 46: Preprocessing block diagram in UNIQUE.

5.4.1.1 Color Space Selection

The human visual system is more sensitive to changes in intensity compared to color as exploited in the chroma subsampling for image coding [27]. Therefore, luma channels can be more informative compared to chroma channels in terms of perceived quality. Although color may not be as informative as intensity, there is still additional information in color, that is not conveyed by intensity. An intuitive way to introduce color information is to directly use RGB channels. However, there is a high correlation between color channels of RGB images: about 0.78 for r_{BR} (cross correlation between the B and the R channels), 0.98 for r_{GR}, and 0.94 for r_{GB} [124]. Based on these correlations, the G channel has most of the information already contained in the R and the B channels so we use the G channel in the introduced method. Chroma and luma information can be seperated by transforming RGB images into YCbCr images. In addition to the G channel, we also use the luminance channel

because it includes structural information. The Cr channel is also used in the introduced method, based on promising results in the experimentation. We combine these three most descriptive channels to construct a YGCr image.

5.4.1.2 Patch-based Sampling and Reshaping

An 8×8 patch is randomly sampled from an input image and converted into a 1-D vector of length 64 either in row major or column major order. Three channels are concatenated into a 1-D vector of length 192.

5.4.1.3 Mean Subtraction and ZCA Whitening

A color space selection procedure is performed to reduce data redundancy between color channels. However, adjacent pixel values in any single channel are also highly correlated. The authors in [125] show that retina and lateral geniculate nucleus in a HVS perform whitening [126] for decorrelation. In the introduced method, we perform a ZCA whitening operation. For each location in reshaped vectors, we compute a mean value over all patches and perform mean subtraction. Then, we perform a patch-wise ZCA whitening operation to decorrelate spatial representations.

5.4.2 Unsupervised Image Quality Estimation

5.4.2.1 Training Set

Learning-based image quality assessment methods usually perform training using image quality databases or simulated distortions, which can limit the image quality assessment capability to specific distortions rather than general scenarios. Therefore, to avoid overfitting, we use the ImageNet database. In total, the database includes around 14 million images according to the update on April 30, 2010. Images can contain a queried object along with other objects, multiple instances, occlusion or text [127]. In our training phase, we randomly select around $1,000$ images and extract 100 patches from each image, which leads to a total of $100,000$ patches.

5.4.2.2 Sparse Representation

The authors in [128] claim that sparse coding, with an overcomplete basis set, operates similar to encoding mechanisms of visual representations in a V1 cortex and response characteristics of simple V1 cells can be simulated by learning weight parameters over thousands of patches. In the introduced work, we use a linear decoder architecture to obtain sparse representations.

5.4.2.3 Unsupervised Learning

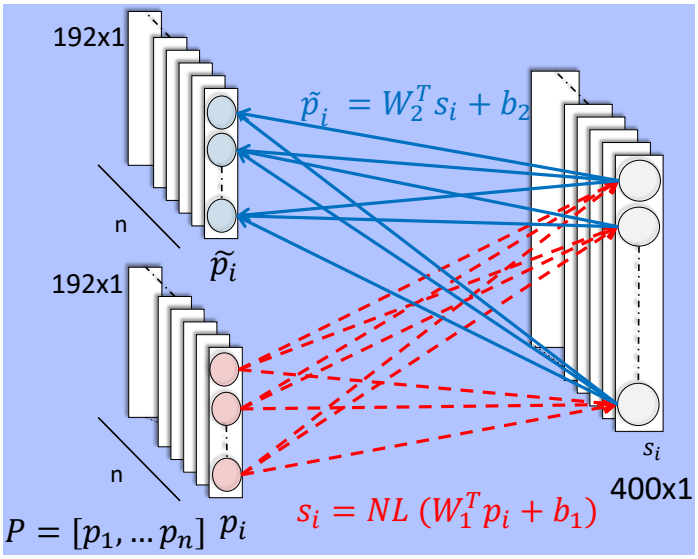


Figure 47: Linear decoder architecture in UNIQUE.

A linear decoder is an unsupervised learning architecture used to represent input data in different dimensions. It is a specific type of an autoencoder, which is based on artificial neural networks. The output of a linear decoder is a reconstructed version of the input and a backpropagation operation is performed to reduce the reconstruction error by adjusting weights and bias. If the target dimension is lower than the original, the linear decoder learns a compact representation similar to PCA. When we set the target dimension higher and add constraints, a sparse representation can be obtained. In the introduced method, we set the target dimension higher than the input and add

a sparsity constraint. The sparse representation s is obtained using the whitened map P as

$$s = W_1^T P + b_1, \tag{73}$$

where W_1 and b_1 refer to a weight matrix and a bias vector, respectively. s is passed through a non-linear sigmoid activation function. The objective function $J(W, b)$ is minimized using a backpropogation operation, in which W includes reconstruction weights W_2 and bias b_2 in addition to W_1 and b_1. Adding a sparsity penalty term weighted by β, the objective function is expressed as

$$J(W, b) = \|(W_2^T s + b_2) - P\|_2^2 + \beta \sum_{j=1}^{N} \mathrm{KL}(\rho||\hat{\rho}_j) + \lambda \|W\|_2^2, \tag{74}$$

where the first term is the reconstruction error, the second term is the sparsity penalty, the third term is the weight decay, KL is the Kullback-Leibler divergence, β is the weight of the sparsity penalty term, N is the number of hidden units, ρ is the target average activation, $\hat{\rho}$ is the actual average activation, and λ is the weight of the weight decay term. Note that s is a function of W_1 and b_1 as in Eq. (73) and minimization is performed over weights and bias in both encoding and decoding stages. The weight decay term corresponds to regularization, which limits weights and prevents overfitting to only particular input units. Minimization is carried out using limited memory Broyden-Fletcher-Goldfarb-Shanno (L-BFGS) algorithm. To account for non-smoothness, we use KL-divergence, which can be expressed as

$$\sum_{j=1}^{N} \mathrm{KL}(\rho||\hat{\rho}_j) = \sum_{j=1}^{N} \rho \log \frac{\rho}{\hat{\rho}_j} + (1-\rho) \log \frac{1-\rho}{1-\hat{\rho}_j}, \tag{75}$$

where N is the number of hidden units, ρ is the target average activation (set to 0.035), and $\hat{\rho}$ is the actual average activation formulated as

$$\hat{\rho} = \frac{1}{M} \sum_{i=1}^{M} s_i, \tag{76}$$

where M is the number of training examples in one forward pass. This type of objective function definition does not only lead to smoothening but also preserves sparsity in the hidden units [129]. β is set to 5 based on empirical studies to control

the weight of the sparsity penalty term. The linear decoder architecture is shown in Fig. 47. An image patch of size 192×1 is mapped to a sparse representation of size 400×1. The nodes in s_i represent hidden layer units.

5.4.2.4 Backpropagation in Unsupervised Learning

The core of the unsupervised learning procedure in the introduced method `UNIQUE` is based on a backpropagation operation. Therefore, to understand the mechanisms of the introduced method, we need to understand the backpropagation operation. In this section, we provide a brief description of backpropagation in a single layer that does not include any non-linearity. However, a similar derivation approach can also be used for systems with non-linearities and multiple layers.

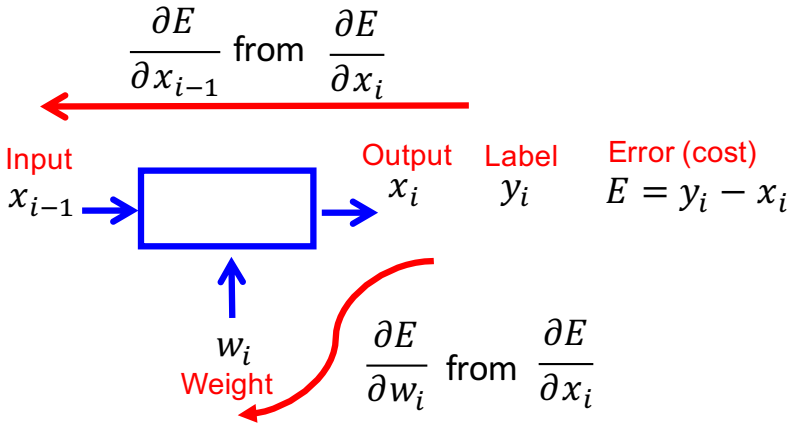


Figure 48: Layer-wise backpropagation computation in a network.

We show the terms and the formulations used in a backpropagation operation in Fig. 48. Let's denote the input of a layer indexed with i as x_{i-1} and output as x_i. The ground truth or label is y_i and the error is $E = y_i - x_i$. We can define system block as a function f_i, whose output depends on x_{i-1} and w_i. We use partial derivatives and chain rule to compute weight updates. At first, we take the partial derivative of the error term with respect to the weight using the chain rule as

$$\frac{\partial E}{\partial w_i} = \frac{\partial E}{\partial x_i} \cdot \frac{\partial f_i(x_{i-1}, w_i)}{\partial w_i}, \tag{77}$$

where ∂ is the partial derivative operator, x_{i-1} is the input, w_i is the weight, x_i is the output, and $f_i(x_{i-1}, w_i)$ is the system function corresponding to the block indexed with i. The system function can be expressed as

$$f_i(x_{i-1}, w_i) = w_i^T \cdot x_{i-1}, \tag{78}$$

where w_i^T is the transpose of the weight. The partial derivative of the system function f_i with respect to the weight w_i is

$$\frac{\partial f(x_{i-1}, w_i)}{\partial w_i} = \frac{\partial (w_i^T \cdot x_{i-1})}{\partial w_i} = x_{i-1}. \tag{79}$$

Partial derivative of the error with respect to the weight given in Eq. (77) can be directly computed using the partial derivative of the error with respect to the output, Eq. (78), and Eq. (79). Given the partial derivative of the error with respect to the weight, we can update the weight in the i^{th} layer as

$$w_i = w_i - \eta \frac{\partial E}{\partial w_i}, \tag{80}$$

where η corresponds to the learning rate of the update.

5.4.2.5 Visualization of Weights and Feature Maps

In Fig. 49, we show the weights that contain edge, orientation, and color information learned by individual hidden layer units. Each square partition shows the patch that maximally activates hidden units. A weighted sum of these hidden units is used to approximate natural images. In Fig. 50, we represent a reference and a distorted image using the visual patterns that maximally activate hidden units. UNIQUE features capture the perceived differences between the reference and the distorted images, which can be summarized as follows: When we compare the reference and the distorted images, the degradations over the sky region are obvious in the top grids and also in the right side of the middle row. There is a significant color degradation in the upper part of the top row. Degradations are less observable around the highly

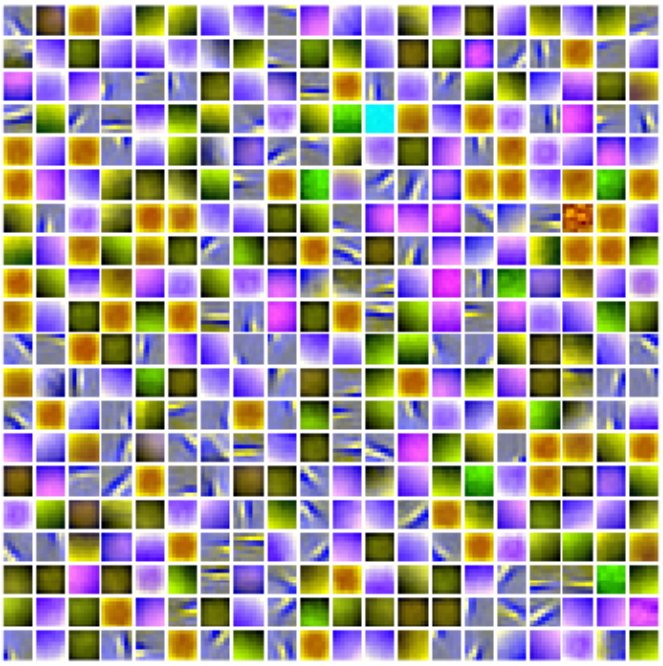


Figure 49: Visualization of learned weights in UNIQUE.

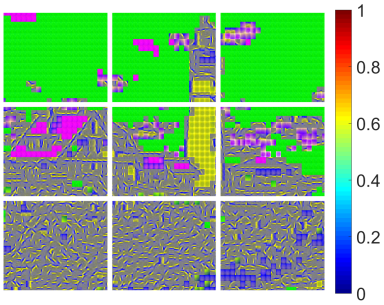
textured regions as observed in the bottom grids, where we have the textured rocks. In the middle row, we can see degradation in the roof of the houses and around the windows, where we have edges or sharp transitions. However, it is not easy to observe degradations in regions with over exposure such as the surface of the lighthouse. The regions that correspond to perceivable degradations are generally represented differently in the UNIQUE maps, which correspond the reference and the distorted images.

5.4.2.6 Image Quality Estimation

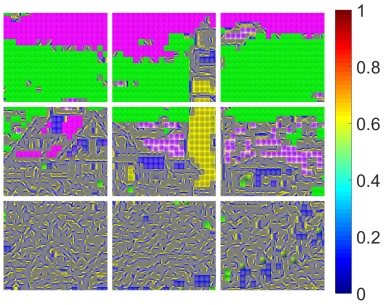
The processes that lead to an estimated quality score, using reference and distorted images, are shown in Fig. 51. A linear decoder is trained in an unsupervised fashion by feeding preprocessed patches. The learned weights and the bias are used along with a non-linear mapping to transform preprocessed non-overlapping patches of the reference and the compared images into sparse representations. These representations

(a)A reference image.

(b) A distorted image.



(c)UNIQUE map of a reference image.



(d) UNIQUE map of a distorted image.

Figure 50: UNIQUE maps corresponding to reference and distorted images.

are reshaped into vectors. If an entity in these vectors is significantly less than the average activation value, a zero is assigned to mimic suppression mechanisms in a visual system. The Spearman rank order correlation coefficient is used to compare two reshaped vectors and we use 10^{th} power of the correlation coefficient to utilize full quality estimate range.

Contrary to sparse coding approaches, we do not approximate natural image patches using overcomplete basis. We treat learned weights as filters and use inner product to calculate the response of each patch to every filter. In effect, we project input patches on the filters and obtain sparse representations based on their responses, which are hidden unit activations. These sparse representations are suppressed and compared to obtain image quality scores. Sparse coding-based methods generally perform optimization even during the testing stage, which is not performed in the introduced method. Because of this inherent difference, `UNIQUE` is advantageous

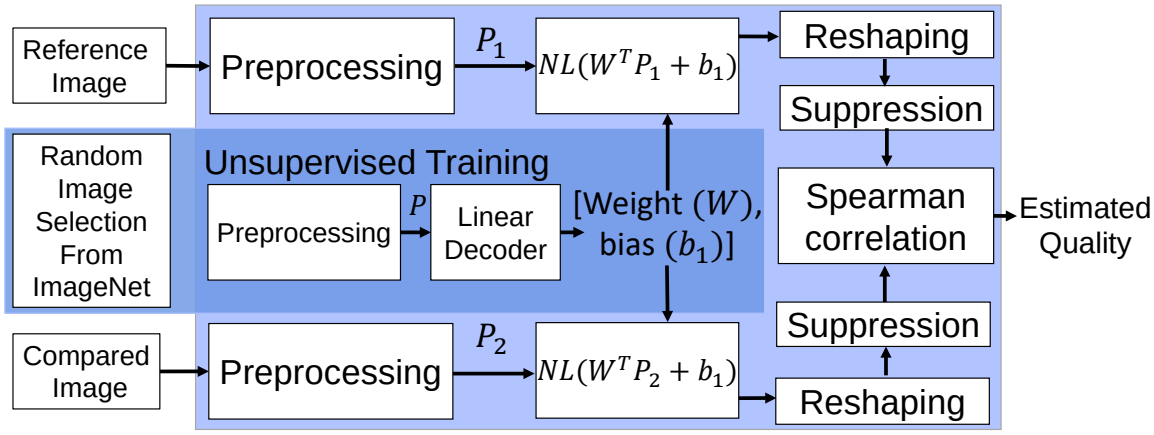


Figure 51: Image quality estimation block diagram in UNIQUE.

in terms of computational complexity.

5.4.3 Wide and Deep Extensions of UNIQUE

5.4.3.1 MS-UNIQUE: Multi-model and Sharpness-weighted Unsupervised Image Quality Estimation

We build on `UNIQUE` by weighting the patterns learned by individual hidden layer units with the descriptiveness of these representations. The descriptiveness is measured by Kurtosis, which is the forth standardized moment. Kurtosis is used as a shape descriptor of probability distributions and it is usually known as a measure of sharpness, peakedness, or tailedness of a distribution. In this work, we use the sharpness description and denote the weighted representations as sharpness-weighted.

In Fig. 52, we show two weight sets that are separated through a Kurtosis-based sharpness threshold. Weight sets with high Kurtosis capture edges and sharp transitions with orientation information whereas weight sets with low Kurtosis capture color information and smoother transitions with orientation information. Therefore, sharpness weights are higher for weight sets that contain more structural information. We scale weight sets to mimic the sensitivity of a visual system. As exploited in the chroma subsampling for image coding [27], a human visual system is more sensitive

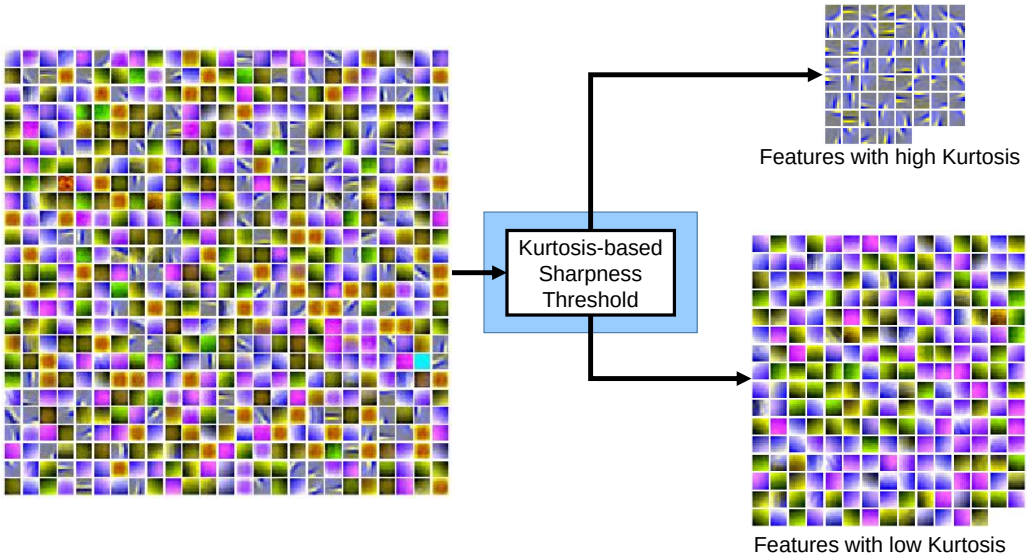


Figure 52: Classification of the learned representations based on Kurtosis in MS-UNIQUE.

to changes in structure compared to color.

Moreover, we also use multiple decoders with different number of hidden layers to represent image patches through different abstraction levels. In `UNIQUE`, linear decoders are used to obtain representations that are dimensionally greater than the input representation. However, in `MS-UNIQUE`, compact representations are used along with sparse representations while maintaining the sparsity criterion. The feature generation block diagram of `MS-UNIQUE` is given in Fig. 53. We use the `UNIQUE` pipeline with different number of hidden units, which can be sorted as 81, 121, 169, 400, and 625. The weight sets learned by different models are weighted based on their Kurtosis values. Finally, the output of each model is concatenated to obtain a single representation corresponding to an input image. Visual representations are compared to obtain a quality estimate as in `UNIQUE`.

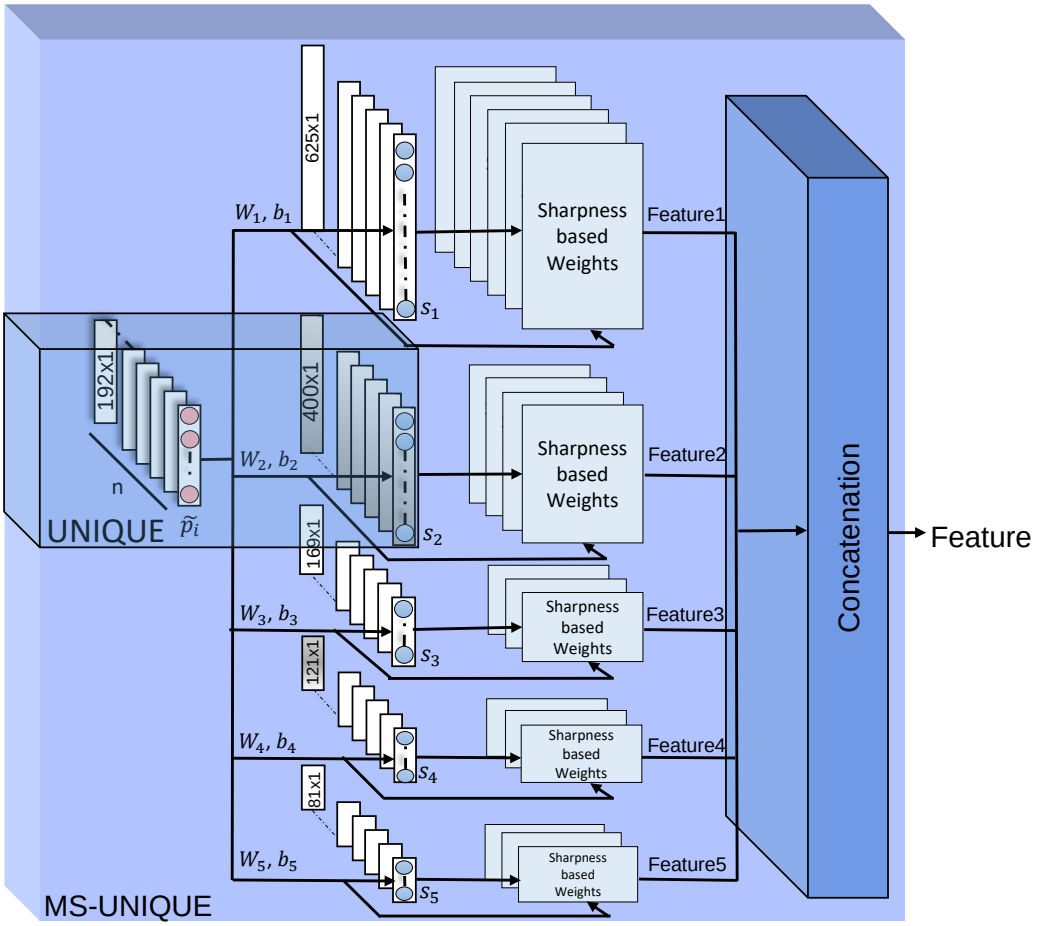


Figure 53: Feature generation block diagram in MS-UNIQUE.

5.4.3.2 DMS-UNIQUE: Deep Multi-model and Sharpness-weighted Unsupervised Image Quality Estimation

As in `UNIQUE` and `MS-UNIQUE`, we project input patches on learned weight sets to obtain abstract representations. In `MS-UNIQUE`, we directly concatenate these representations obtained from different models. On contrary, in `DMS-UNIQUE`, we select the highest 40 responses of each model after projecting input patches on sharpness-weighted weight sets. The selection process among the projected input patches is similar to max pooling, in which we select the highest 40 projections rather than a single one. We combine maximum 40 projections from 5 different models to obtain a representation of length 200. To further increase the abstraction level, we process

the concatenated visual representation as in UNIQUE. We normalize the concatenated visual representation with a mean subtraction and a ZCA whitening operation. The normalized representation is processed with multiple linear decoders, which have 169, 256, and 121 hidden units. The weight sets in these models are also weighted with sharpness (Kurtosis). The output of these linear decoders are concatenated with the output of the UNIQUE representation. The concatenated representations corresponding to compared images are thresholded to mimic suppression mechanisms in a visual system and 10^{th} power of the correlation coefficient between suppressed representations is computed to obtain a quality estimate.

5.4.3.3 D-UNIQUE: Deep Unsupervised Image Quality Estimation

Instead of using a single linear decoder, we can stack linear decoders to obtain higher levels of abstractions. We train the stacked linear decoder architecture in a greedy layer-wise fashion. At first, we train the first layer with preprocessed patches. Then, the trained linear decoder in the first layer is used to obtain sparse representations that are fed to the next layer. The output of each layer is used as the input for the next layer. We tune the parameters of each layer individually as shown in Fig. 54.

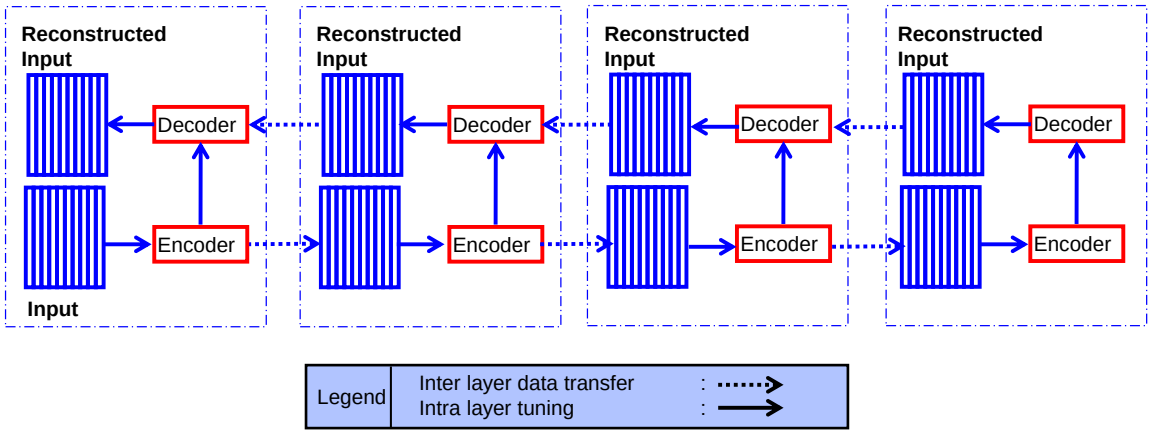


Figure 54: Feature generation block diagram in D-UNIQUE.

In a deep network, we use back propagation to propagate the error layer by layer and adjust the weights accordingly. The equations described in Section 5.4.2.4 are

used to update the weight of a single layer. We need to backpropagate the error to previous layers to update the previous weights as well. We calculate the partial derivative of error with respect to the weight of the system in the previous layer as

$$\frac{\partial E}{\partial w_{i-1}} = \frac{\partial E}{\partial x_{i-1}} \cdot \frac{\partial f_i(x_{i-2}, w_{i-1})}{\partial w_{i-1}}, \tag{81}$$

where we use the chain rule to split the formulation into the partial derivative of the error with respect to the output of the previous layer times the partial derivative of the system function in the previous layer with respect to the weight in the previous layer. The first term that corresponds to the partial derivative of the error with respect to the output of the previous layer can be calculated as

$$\frac{\partial E}{\partial x_{i-1}} = \frac{\partial E}{\partial x_i} \cdot \frac{\partial f_i(x_{i-1}, w_i)}{\partial x_{i-1}}, \tag{82}$$

where the partial derivative of the error with respect to the output of i^{th} layer is multiplied by the partial derivative of the the system function of the i^{th} layer with respect to the output of the previous layer. The second term can be calculated as

$$\frac{\partial f_i(w_i^T \cdot x_{i-1})}{\partial x_{i-1}} = w_i^T, \tag{83}$$

where the partial derivative leads to the transpose of the weights in the i^{th} layer. Given Eq. (83) and the partial derivative of the error with respect to the output of the i^{th} layer, we can directly calculate Eq. (82). Given Eq. (82) and Eq. (79) with an index $i-1$ instead of i, we can calculate Eq. (81), which is used to update the weight in the previous layer. We can follow these steps to update all the weights in each layer.

When we extend `UNIQUE` by solely stacking linear decoders, the patterns learned by the weight sets remain similar to the patterns learned by `UNIQUE`. Moreover, the estimation performance do not change significantly. In `UNIQUE`, we obtain sparse representations corresponding to local patches of size 8×8. Therefore, in the case of solely stacking linear decoders for small patches, even a single layer is able to learn

descriptive transformations and deeper architectures lead to similar representational power. We do not explicitly include `D-UNIQUE` in the performance evaluation or as a method contributed to the literature because of its similarity to `UNIQUE` in terms of performance and operation mechanisms.

5.5 Performance Evaluation of Image Quality Estimators

5.5.1 Outlier Ratio, Root Mean Square Error, and Correlation

The performance of introduced and existing image quality estimators are reported in Table 12 in terms of outlier ratio, root mean square error, the Pearson correlation, and the Spearman correlation. In each category, we highlight three best performing image quality assessment algorithms with a bold typeset. We highlight more than three methods when they lead to equivalent performances. At least one of the introduced quality estimators is always among the top performing quality estimators. `UNIQUE` is among the top performing quality estimators in all the categories and databases other than the Spearman correlation in the LIVE database. `CSV` is among the top performing methods in terms of outlier ratio and Pearson in the MULTI database, in terms of root mean square error in all the databases, and in the LIVE database in both correlation categories. `PerSIM` is among the top performing methods in terms of root mean square error and the Pearson in the MULTI and the TID13 databases, and in terms of the Spearman correlation in the TID13 database. SR-SIM is the only existing method that is consistently among the top performing quality estimators.

Table 12: Overall performance of image quality estimators.

	Existing Methods													Proposed Methods		
Methods	**PSNR**	**PSNR HA** [19]	**PSNR HMA** [19]	**SSIM** [130]	**MS SSIM** [14]	**CW SSIM** [45]	**IW SSIM** [16]	**SR SIM** [35]	**FSIM** [11]	**FSIM c** [11]	**BRIS QUE** [21]	**BIQI** [131]	**BLII NDS2** [132]	**Per SIM** [1]	**CSV** [2]	**UNI QUE** [4]
	Outlier Ratio															
MULTI	0.008	0.013	0.008	0.015	0.013	0.093	0.013	**0**	0.017	0.015	0.066	0.024	0.077	0.004	**0**	**0**
TID13	0.725	**0.615**	0.670	0.733	0.691	0.855	0.700	**0.632**	0.741	0.727	0.851	0.855	0.851	0.655	0.687	**0.641**
	Root Mean Square Error															
LIVE	8.61	6.93	**6.58**	7.52	7.44	11.2	7.11	7.54	7.28	7.20	8.57	10.8	9.04	6.80	**5.84**	**6.76**
MULTI	12.7	11.3	10.7	11.0	11.2	18.8	10.0	**8.68**	10.8	10.7	15.0	12.7	17.4	**9.89**	**9.89**	**9.25**
TID13	0.87	0.65	0.69	0.76	0.69	1.20	0.68	**0.61**	0.71	0.68	1.10	1.10	1.09	**0.64**	**0.64**	**0.61**
	Pearson Correlation Coefficient															
LIVE	0.928	0.953	**0.958**	0.945	0.946	0.872	0.951	0.945	0.949	0.950	0.928	0.883	0.920	0.955	**0.967**	**0.956**
MULTI	0.739	0.801	0.821	0.812	0.802	0.379	0.847	**0.888**	0.818	0.821	0.605	0.738	0.389	**0.852**	**0.852**	**0.872**
TID13	0.705	0.850	0.827	0.788	0.830	0.227	0.831	**0.866**	0.819	0.832	0.460	0.448	0.473	**0.854**	0.853	**0.868**
	Spearman Correlation Coefficient															
LIVE	0.909	0.937	0.944	0.949	0.951	0.902	**0.960**	0.955	**0.961**	**0.959**	0.939	0.897	0.922	0.950	**0.959**	0.952
MULTI	0.677	0.714	0.743	0.860	0.836	0.630	**0.883**	**0.866**	0.863	**0.866**	0.598	0.610	0.386	0.818	0.848	**0.866**
TID13	0.700	0.847	0.817	0.741	0.785	0.562	0.777	0.807	0.801	**0.851**	0.414	0.393	0.396	**0.853**	0.845	**0.860**

In addition to the performance of image quality assessment algorithms on overall databases, we also report the performance in individual distortion types in Tables 13, 14, 15, and 16. In each category, we highlight three best performing image quality assessment algorithms with a bold typeset. We highlight more than three methods when they lead to equivalent performances. In Table 13, we report the outlier ratios for MULTI and TID13 because standard deviations of subjective scores are not reported in the LIVE database. At least one of the introduced quality estimators is among the top performing methods other than the blur category in the TID13 database. Out of 7 distortion types (compression, noise, communication, blur, color, global, and local), the existing methods that are among the top performing methods are PSNR in 1 type (local), PSNR-HA in 6 types (compression, noise, communication, blur, color, and global), PSNR-HMA in 2 types (compression and blur), MS-SSIM in 2 types (communication and global), and SR-SIM in 4 types (compression, noise, blur, and local). The introduced image quality estimators that are among the top performing methods are `PerSIM` in 4 types (compression, noise, blur, and global), `CSV` in 4 types (compression, noise, blur, and color), and `UNQIUE` in 6 types (compression, noise, communication, blur, color, and local).

We report the root mean square error performances in Table 14. Out of 7 distortion types, the existing image quality estimators that are among the top performing methods are PSNR-HA in 5 types (noise, communication, blur, color, and global), PSNR-HMA in 3 types (compression, noise, and communication), SSIM in 2 types (compression and communication), MS-SSIM in 3 types (communication, global, and local), IW-SSIM in 2 types (compression and blur), SR-SIM in 5 types (compression, noise, communication, blur, and local), and FSIMc in 2 types (compression and blur). The introduced image quality estimators that are among the top performing methods are `PerSIM` in 3 types (compression, noise, and blur), `CSV` in 5 types (compression, noise, communication, blur, and color), and `UNIQUE` in 6 types (compression, noise,

Table 13: Outlier ratio performance of image quality estimators over degradation categories.

	Existing Methods								
Distortion Types	**Databases**	**PSNR**	**PSNR HA** [19]	**PSNR HMA** [19]	**SSIM** [130]	**MS SSIM** [14]	**CW SSIM** [45]	**IW SSIM** [16]	**SR SIM** [35]
Comp.	**Compression [TID13]**	0.698	**0.544**	0.594	0.704	0.664	0.898	0.682	**0.592**
	Blur-Jpeg [MULTI]	0.008	0.004	**0**	0.004	0.004	0.093	0.004	**0**
Noise	**Blur-Noise [MULTI]**	0.008	0.022	0.017	0.026	0.022	0.093	0.022	**0**
	Noise [TID13]	0.696	**0.571**	0.633	0.698	0.680	0.853	0.679	**0.578**
Comm.	**Communication [TID13]**	0.860	**0.680**	0.700	0.716	**0.676**	0.864	0.712	0.716
Blur	**Blur [TID13]**	0.700	**0.580**	**0.608**	0.708	0.712	0.880	0.676	**0.600**
	Blur-Jpeg [MULTI]	0.008	0.004	**0**	0.004	0.004	0.093	0.004	**0**
	Blur-Noise [MULTI]	0.008	0.022	0.017	0.026	0.022	0.093	0.022	**0**
Color	**Color [TID13]**	0.688	**0.672**	0.704	0.746	0.709	0.840	0.736	0.736
Global	**Global [TID13]**	0.820	**0.608**	0.684	0.804	**0.620**	0.888	0.676	0.676
Local	**Local [TID13]**	**0.712**	0.808	0.872	0.880	0.808	0.780	0.776	**0.680**
	Existing Methods						**Proposed Methods**		
Distortion Types	**Databases**	**FSIM** [11]	**FSIMc** [11]	**BRISQUE** [21]	**BIQI** [131]	**BLIINDS2** [132]	**PerSIM** [1]	**CSV** [2]	**UNIQUE** [4]
Comp.	**Compression [TID13]**	0.725	0.722	0.864	0.866	0.880	**0.586**	0.642	0.608
	Blur-Jpeg [MULTI]	0.004	0.004	0.008	0.017	0.053	**0**	**0**	**0**
Noise	**Blur-Noise [MULTI]**	0.031	0.026	0.124	0.031	0.102	0.008	**0**	**0**
	Noise [TID13]	0.700	0.678	0.826	0.839	0.823	**0.581**	0.635	0.604
Comm.	**Communication [TID13]**	0.748	0.744	0.856	0.880	0.868	0.796	0.828	**0.676**
Blur	**Blur [TID13]**	0.792	0.760	0.852	0.840	0.852	0.640	0.764	0.716
	Blur-Jpeg [MULTI]	0.004	0.004	0.008	0.017	0.053	**0**	**0**	**0**
	Blur-Noise [MULTI]	0.031	0.026	0.1244	0.031	0.102	0.008	**0**	**0**
Color	**Color [TID13]**	0.784	0.789	0.861	0.840	0.866	0.728	**0.674**	**0.637**
Global	**Global [TID13]**	0.732	0.724	0.868	0.880	0.852	**0.656**	0.700	0.704
Local	**Local [TID13]**	0.848	0.836	0.904	0.900	0.908	0.840	0.784	**0.708**

blur, color, global, and local).

We report the Pearson correlation performances in Table 15. Out of 7 distortion types, the existing image quality estimators that are among the top performing methods are PSNR in 1 type (color), PSNR-HA in 3 types (noise, blur, and global), PSNR-HMA in 6 types (compression, noise, communication, blur, global, and local), SSIM in 2 types (compression and communication), MS-SSIM in 2 types (communication and global), IW-SSIM in 2 types (compression and blur), SR-SIM in 5 types (compression, noise, communication, blur, and local), and FSIMc in 1 type (compression). The introduced image quality estimators that are among the top performing methods are `PerSIM` in 3 types (compression, noise, and blur), `CSV` in 6 types (compression,

Table 14: Root mean square error performance of image quality estimators over degradation categories.

	Existing Methods								
Distortion Types	**Databases**	**PSNR**	**PSNR HA** [19]	**PSNR HMA** [19]	**SSIM** [130]	**MS SSIM** [14]	**CW SSIM** [45]	**IW SSIM** [16]	**SR SIM** [35]
Comp.	**Jp2k [LIVE]**	9.08	7.74	7.36	**6.57**	6.99	14.4	**6.66**	7.49
	Jpeg [LIVE]	8.77	6.70	**6.20**	7.60	7.55	11.6	7.17	7.66
	Compression [TID13]	0.85	0.56	0.59	0.62	0.54	1.48	0.55	**0.44**
	Blur-Jpeg [MULTI]	13.4	11.2	10.4	11.6	12.0	19.1	10.4	**8.18**
Noise	**Wn [LIVE]**	9.19	7.03	**6.65**	7.17	6.99	9.55	7.35	7.74
	Blur-Noise [MULTI]	12.0	11.4	11.1	10.37	10.48	18.59	9.58	**9.16**
	Noise [TID13]	0.67	**0.46**	0.50	0.59	0.58	1.01	0.57	**0.47**
Comm.	**FF [LIVE]**	7.69	**5.98**	**5.38**	8.65	8.38	9.25	6.95	6.33
	Communication [TID13]	0.77	0.74	0.76	**0.57**	**0.57**	1.26	0.60	**0.54**
Blur	**GBlur [LIVE]**	8.00	**6.90**	6.98	7.71	7.27	9.43	7.49	8.32
	Blur [TID13]	1.02	0.67	0.66	0.71	0.64	1.66	**0.59**	**0.48**
	Blur-Jpeg [MULTI]	13.4	11.2	10.4	11.6	12.0	19.1	10.4	**8.1**
	Blur-Noise [MULTI]	12.0	11.4	11.1	10.3	10.4	18.5	9.58	**9.16**
Color	**Color [TID13]**	0.67	**0.64**	0.68	0.94	0.98	1.10	0.95	0.95
Global	**Global [TID13]**	1.65	**0.74**	0.87	1.05	**0.85**	1.40	0.86	0.89
Local	**Local [TID13]**	0.83	1.14	1.21	1.13	**0.78**	0.95	0.81	**0.63**
	Existing Methods						**Proposed Methods**		
Distortion Types	**Databases**	**FSIM** [11]	**FSIM c** [11]	**BRISQUE** [21]	**BIQI** [131]	**BLIINDS2** [132]	**Per SIM** [1]	**CSV** [2]	**UNIQUE** [4]
Comp.	**Jp2k [LIVE]**	6.78	**6.64**	8.99	12.9	9.78	7.13	**6.66**	6.89
	Jpeg [LIVE]	7.35	7.32	8.27	10.3	9.17	**5.55**	**5.88**	6.73
	Compression [TID13]	0.59	0.57	1.06	1.18	1.18	**0.47**	**0.48**	0.50
	Blur-Jpeg [MULTI]	10.9	10.9	12.2	12.5	16.0	**9.62**	10.54	**9.65**
Noise	**Wn [LIVE]**	7.56	7.38	8.61	10.4	8.14	6.96	**5.65**	**6.45**
	Blur-Noise [MULTI]	10.7	10.5	17.3	12.8	18.7	10.1	**9.20**	**8.83**
	Noise [TID13]	0.58	0.56	0.96	0.99	0.98	**0.45**	0.55	0.53
Comm.	**FF [LIVE]**	6.60	6.97	7.20	9.32	7.51	7.25	**5.69**	7.64
	Communication [TID13]	0.78	0.77	1.31	1.29	1.23	0.70	0.69	0.58
Blur	**GBlur [LIVE]**	8.12	7.77	9.55	10.4	10.0	7.25	**4.89**	**5.97**
	Blur [TID13]	0.61	**0.60**	1.32	1.27	1.33	**0.60**	0.66	0.66
	Blur-Jpeg [MULTI]	10.9	10.9	12.2	12.5	16.0	**9.62**	10.5	**9.65**
	Blur-Noise [MULTI]	10.7	10.5	17.3	12.8	18.7	10.1	**9.20**	**8.83**
Color	**Color [TID13]**	0.89	0.81	1.09	1.04	1.02	0.73	**0.61**	**0.64**
Global	**Global [TID13]**	0.89	0.88	1.24	1.23	1.20	0.94	0.94	**0.84**
Local	**Local [TID13]**	0.87	0.88	1.16	1.15	1.13	0.99	0.80	**0.74**

noise, communication, blur, color, and local), and `UNIQUE` in 5 types (compression, noise, communication, blur, and color).

We report the Spearman correlation performances in Table 16. Out of 7 distortion types, the existing image quality estimators that are among the top performing methods are PSNR in 2 types (communication and color), PSNR-HA in 3 types

Table 15: Pearson correlation performance of image quality estimators over degradation categories.

	Existing Methods								
Distortion Types	Databases	PSNR	PSNR HA [19]	PSNR HMA [19]	SSIM [130]	MS SSIM [14]	CW SSIM [45]	IW SSIM [16]	SR SIM [35]
Comp.	Jp2k [LIVE]	0.912	0.943	0.951	**0.953**	0.947	0.810	**0.953**	0.942
	Jpeg [LIVE]	0.932	0.960	0.966	0.953	0.954	0.872	0.958	0.950
	Compression [TID13]	0.901	0.964	**0.969**	0.939	0.947	0.448	0.944	**0.970**
	Blur-Jpeg [MULTI]	0.722	0.810	0.838	0.797	0.792	0.408	0.838	**0.904**
Noise	Wn [LIVE]	0.942	**0.971**	**0.975**	0.962	0.964	0.929	0.960	0.957
	Blur-Noise [MULTI]	0.773	0.791	0.804	0.833	0.838	0.375	0.858	**0.871**
	Noise [TID13]	0.776	**0.916**	**0.907**	0.837	0.843	0.464	0.839	0.894
Comm.	FF [LIVE]	0.972	0.980	**0.985**	0.976	0.977	0.949	0.981	0.974
	Communication [TID13]	0.766	0.813	0.800	**0.895**	**0.889**	0.139	0.869	**0.905**
Blur	GBlur [LIVE]	0.952	**0.972**	**0.972**	0.955	0.958	0.920	0.956	0.947
	Blur [TID13]	0.865	0.922	0.931	0.935	0.946	0.254	**0.948**	**0.959**
	Blur-Jpeg [MULTI]	0.722	0.810	0.838	0.797	0.792	0.408	0.838	**0.904**
	Blur-Noise [MULTI]	0.773	0.791	0.804	0.833	0.838	0.375	0.858	**0.871**
Color	Color [TID13]	**0.844**	0.841	0.814	0.670	0.666	0.355	0.675	0.672
Global	Global [TID13]	0.342	**0.744**	**0.762**	0.560	**0.656**	0.653	0.652	0.571
Local	Local [TID13]	0.647	0.687	**0.732**	0.235	0.669	0.375	0.698	**0.831**
	Existing Methods						Proposed Methods		
Distortion Types	Databases	FSIM [11]	FSIMc [11]	BRISQUE [21]	BIQI [131]	BLIINDS2 [132]	PerSIM [1]	CSV [2]	UNIQUE [4]
Comp.	Jp2k [LIVE]	0.951	**0.953**	0.914	0.820	0.893	0.948	0.951	0.948
	Jpeg [LIVE]	0.956	0.956	0.946	0.910	0.929	**0.976**	**0.980**	**0.972**
	Compression [TID13]	0.922	0.926	0.801	0.649	0.607	0.965	**0.968**	0.950
	Blur-Jpeg [MULTI]	0.819	0.819	0.854	0.761	0.591	**0.864**	0.836	**0.865**
Noise	Wn [LIVE]	0.959	0.961	0.936	0.908	0.945	0.966	**0.974**	0.967
	Blur-Noise [MULTI]	0.817	0.823	0.448	0.730	0.175	0.839	**0.871**	**0.882**
	Noise [TID13]	0.839	0.847	0.420	0.363	0.369	**0.903**	0.852	0.867
Comm.	FF [LIVE]	0.976	0.977	0.964	0.922	0.941	0.968	**0.986**	**0.983**
	Communication [TID13]	0.824	0.825	0.185	0.218	0.355	0.854	0.836	0.880
Blur	GBlur [LIVE]	0.947	0.951	0.918	0.902	0.918	0.958	**0.979**	0.968
	Blur [TID13]	0.924	0.925	0.775	0.707	0.662	0.935	**0.962**	0.946
	Blur-Jpeg [MULTI]	0.819	0.819	0.854	0.761	0.591	**0.864**	0.836	**0.865**
	Blur-Noise [MULTI]	0.817	0.823	0.448	0.730	0.175	0.839	**0.871**	**0.882**
Color	Color [TID13]	0.673	0.727	0.484	0.447	0.494	0.800	**0.887**	**0.847**
Global	Global [TID13]	0.649	0.646	0.008	0.174	0.027	0.585	0.410	0.510
Local	Local [TID13]	0.701	0.705	0.039	0.057	0.142	0.447	**0.872**	0.721

(compression, noise, and global), PSNR-HMA in 2 types (communication and global), SSIM in 4 types (compression, noise, blur, and global), MS-SSIM in 2 types (communication and blur), IW-SSIM in 3 types (compression, noise, and blur), SR-SIM in 5 types (compression, noise, communication, blur, and local), FSIM in 2 types (compression and blur), and FSIMc in 4 types (compression, communication, blur, and

local). The introduced image quality estimators that are among the top performing methods are `PerSIM` in 4 types (compression, noise, communication, and blur), `CSV` in 5 types (compression, communication, blur, color, and local), and `UNIQUE` in 5 types (compression, noise, communication, blur, and color).

Table 16: Spearman correlation performance of image quality estimators over degradation categories.

	Existing Methods								
Distortion Types	**Databases**	**PSNR**	**PSNR HA** [19]	**PSNR HMA** [19]	**SSIM** [130]	**MS SSIM** [14]	**CW SSIM** [45]	**IW SSIM** [16]	**SR SIM** [35]
Compression	**Jp2k [LIVE]**	0.909	0.935	0.945	**0.971**	0.966	0.868	0.967	0.964
	Jpeg [LIVE]	0.931	0.957	0.966	0.978	0.980	0.925	**0.982**	0.973
	Compression [TID13]	0.910	**0.963**	0.955	0.935	0.940	0.868	0.937	**0.967**
	Blur-Jpeg [MULTI]	0.662	0.706	0.742	0.848	0.839	0.639	0.870	0.862
Noise	**Wn [LIVE]**	0.933	0.959	0.963	0.966	0.967	0.934	0.963	0.954
	Blur-Noise [MULTI]	0.708	0.717	0.738	**0.876**	0.862	0.631	**0.893**	0.863
	Noise [TID13]	0.761	**0.911**	0.894	0.847	0.852	0.802	0.856	**0.900**
Communication	**FF [LIVE]**	**0.991**	0.989	**0.991**	0.982	0.984	0.982	0.980	0.986
	Communication [TID13]	0.773	0.822	0.812	0.894	**0.899**	0.648	0.870	**0.907**
Blur	**GBlur [LIVE]**	0.939	0.960	0.965	**0.969**	**0.969**	0.950	0.967	0.962
	Blur [TID13]	0.878	0.943	0.945	0.922	0.924	0.846	0.932	**0.950**
	Blur-Jpeg [MULTI]	0.662	0.706	0.742	0.848	0.839	0.639	**0.870**	**0.862**
	Blur-Noise [MULTI]	0.708	0.717	0.738	**0.876**	0.862	0.631	**0.893**	0.863
Color	**Color [TID13]**	**0.814**	0.805	0.778	0.234	0.238	0.371	0.233	0.243
Global	**Global [TID13]**	0.340	**0.612**	**0.672**	**0.499**	0.458	0.325	0.452	0.393
Local	**Local [TID13]**	0.542	0.592	0.635	0.288	0.645	0.699	0.611	**0.810**
	Existing Methods						**Proposed Methods**		
Distortion Types	**Databases**	**FSIM** [11]	**FSIM c** [11]	**BRISQUE** [21]	**BIQI** [131]	**BLIINDS2** [132]	**Per SIM** [1]	**CSV** [2]	**UNIQUE** [4]
Compression	**Jp2k [LIVE]**	**0.968**	**0.970**	0.935	0.838	0.897	0.943	0.935	0.937
	Jpeg [LIVE]	0.981	**0.983**	0.962	0.932	0.936	0.975	**0.982**	0.977
	Compression [TID13]	0.960	0.961	0.767	0.605	0.551	**0.964**	0.959	0.939
	Blur-Jpeg [MULTI]	0.854	0.855	0.790	0.663	0.613	0.812	0.840	**0.862**
Noise	**Wn [LIVE]**	0.964	0.965	0.928	0.906	0.931	0.965	0.962	0.965
	Blur-Noise [MULTI]	0.864	0.869	0.470	0.586	0.184	0.818	0.856	**0.873**
	Noise [TID13]	0.887	0.897	0.426	0.376	0.332	**0.924**	0.840	0.863
Communication	**FF [LIVE]**	0.979	0.983	0.987	0.960	0.961	**0.991**	**0.991**	**0.992**
	Communication [TID13]	0.895	**0.899**	0.222	0.367	0.442	0.865	0.846	0.891
Blur	**GBlur [LIVE]**	**0.969**	**0.970**	0.920	0.907	0.904	0.967	**0.969**	0.965
	Blur [TID13]	0.947	0.947	0.771	0.733	0.733	**0.949**	**0.953**	0.928
	Blur-Jpeg [MULTI]	0.854	0.855	0.790	0.663	0.613	0.812	0.840	**0.862**
	Blur-Noise [MULTI]	0.864	0.869	0.470	0.586	0.184	0.818	0.856	**0.873**
Color	**Color [TID13]**	0.240	0.629	0.401	0.328	0.368	0.762	**0.885**	**0.909**
Global	**Global [TID13]**	0.441	0.439	0.035	0.264	0.041	0.408	0.341	0.356
Local	**Local [TID13]**	0.702	**0.705**	0.042	0.047	0.247	0.470	**0.787**	0.645

Overall, there are 4 performance metrics and 7 distortion types, which lead to 28

categories. Out of these 28 categories, the existing image quality estimators that are generally among the top performing methods are SR-SIM in 19, PSNR-HA in 17, PSNR-HMA in 13, IW-SSIM in 7, and FSIMc in 7 categories. The introduced image quality estimators that are among the top performing methods are `PerSIM` in 14, `CSV` in 20, and `UNQIUE` in 22 categories.

5.5.2 Statistical Significance

To measure the significance of the performance differences in terms of correlation, we report the results of statistical significance tests in Table 17. We compare the performance of the existing methods to the introduced methods and report the results as follows: A zero corresponds to statistically similar performance, a one means that the existing method is statistically superior compared to the introduced method, and a minus one indicates that the existing method is statistically inferior compared to the introduced method. For each method comparison, we provide two statistical test results, the first digit is for the Pearson correlation and the second digit is for the Spearman correlation. `PerSIM` is outperformed by other methods only in 3 categories in the LIVE database, 6 in the MULTI database, and none of the categories in the TID13 database out of 26 overall categories. On contrary, `PerSIM` outperforms other methods in 14 categories in the LIVE database, 14 categories in the MULTI database, and 22 categories in the TID13 database out of 26 overall categories. `CSV` is outperformed by other methods only in 2 categories in the MULTI database, 1 category in the TID13 database, and none of the categories in the LIVE database. In contrast, `CSV` outperforms other methods in 22 categories in the LIVE database, 14 categories in the MULTI database, and 22 categories in the TID13 database out of 26 overall categories. `UNIQUE` is outperformed by other methods only in 2 categories in the LIVE database, and none of the categories in the MULTI and the TID13 databases. On contrary, `UNIQUE` outperforms other methods in 14 categories in the LIVE database,

18 categories in the MULTI database, and 23 categories in the TID13 database out of 26 overall categories. PSNR, CW-SSIM, BRISQUE, BIQI, and BLIINDS are outperformed by the introduced methods in all of the performance categories and the databases. PSNR-HA, PSNR-HMA, and MS-SSIM are either statistically inferior or equivalent. SSIM outperforms `PerSIM` in 1 category whereas it is outperformed by `PerSIM` in 3, `CSV` in 4, and `UNIQUE` in 4 categories. IW-SSIM outperforms `PerSIM` in 2, `CSV` in 1, and `UNIQUE` in 1 category whereas it is outperformed by `PerSIM` in 2, `CSV` in 3, and `UNQIUE` in 2 categories. SR-SIM outperforms `PerSIM` and `CSV` in 2 categories whereas it is outperformed by all the introduced methods in 2 categories. FSIM outperforms `PerSIM` in 2 categories and `UNQIUE` in 1 category whereas it is outperformed by `PerSIM` in 2, and by `CSV` and `UNIQUE` in 3 categories. FSIMc outperforms `PerSIM` in 2 categories whereas it is outperformed by `PerSIM` in 1 and by `CSV` and `UNIQUE` in 2 categories. The number of categories in which the introduced methods significantly outperform the existing methods is generally more than the number of categories in which the existing methods outperform the introduced methods.

Table 17: Statistical significance tests of image quality estimation performance.

	Existing Methods												
Methods	**PSNR**	**PSNR HA** [19]	**PSNR HMA** [19]	**SSIM** [130]	**MS SSIM** [14]	**CW SSIM** [45]	**IW SSIM** [16]	**SR SIM** [35]	**FSIM** [11]	**FSIM c** [11]	**BRISQUE** [21]	**BIQI** [131]	**BLIINDS2** [132]
	Introduced Method: PerSIM												
LIVE	-1 -1	0 -1	0 0	-1 0	-1 0	-1 -1	0 1	-1 0	0 1	0 1	-1 -1	-1 -1	-1 -1
MULTI	-1 -1	-1 -1	0 -1	0 1	-1 0	-1 -1	0 1	1 1	0 1	0 1	-1 -1	-1 -1	-1 -1
TID13	-1 -1	0 0	-1 -1	-1 -1	-1 -1	-1 -1	-1 -1	0 -1	-1 -1	-1 0	-1 -1	-1 -1	-1 -1
	Introduced Method: CSV												
LIVE	-1 -1	-1 -1	-1 -1	-1 -1	-1 -1	-1 -1	-1 0	-1 0	-1 0	-1 0	-1 -1	-1 -1	-1 -1
MULTI	-1 -1	-1 -1	0 -1	0 0	-1 0	-1 -1	0 1	1 0	0 0	0 0	-1 -1	-1 -1	-1 -1
TID13	-1 -1	0 0	-1 -1	-1 -1	-1 -1	-1 -1	-1 -1	1 -1	-1 -1	-1 0	-1 -1	-1 -1	-1 -1
	Introduced Method: UNIQUE												
LIVE	-1 -1	0 -1	0 0	-1 0	-1 0	-1 -1	0 1	-1 0	0 1	0 0	-1 -1	-1 -1	-1 -1
MULTI	-1 -1	-1 -1	-1 -1	-1 0	-1 0	-1 -1	0 0	0 0	-1 0	-1 0	-1 -1	-1 -1	-1 -1
TID13	-1 -1	-1 0	-1 -1	-1 -1	-1 -1	-1 -1	-1 -1	0 -1	-1 -1	-1 0	-1 -1	-1 -1	-1 -1

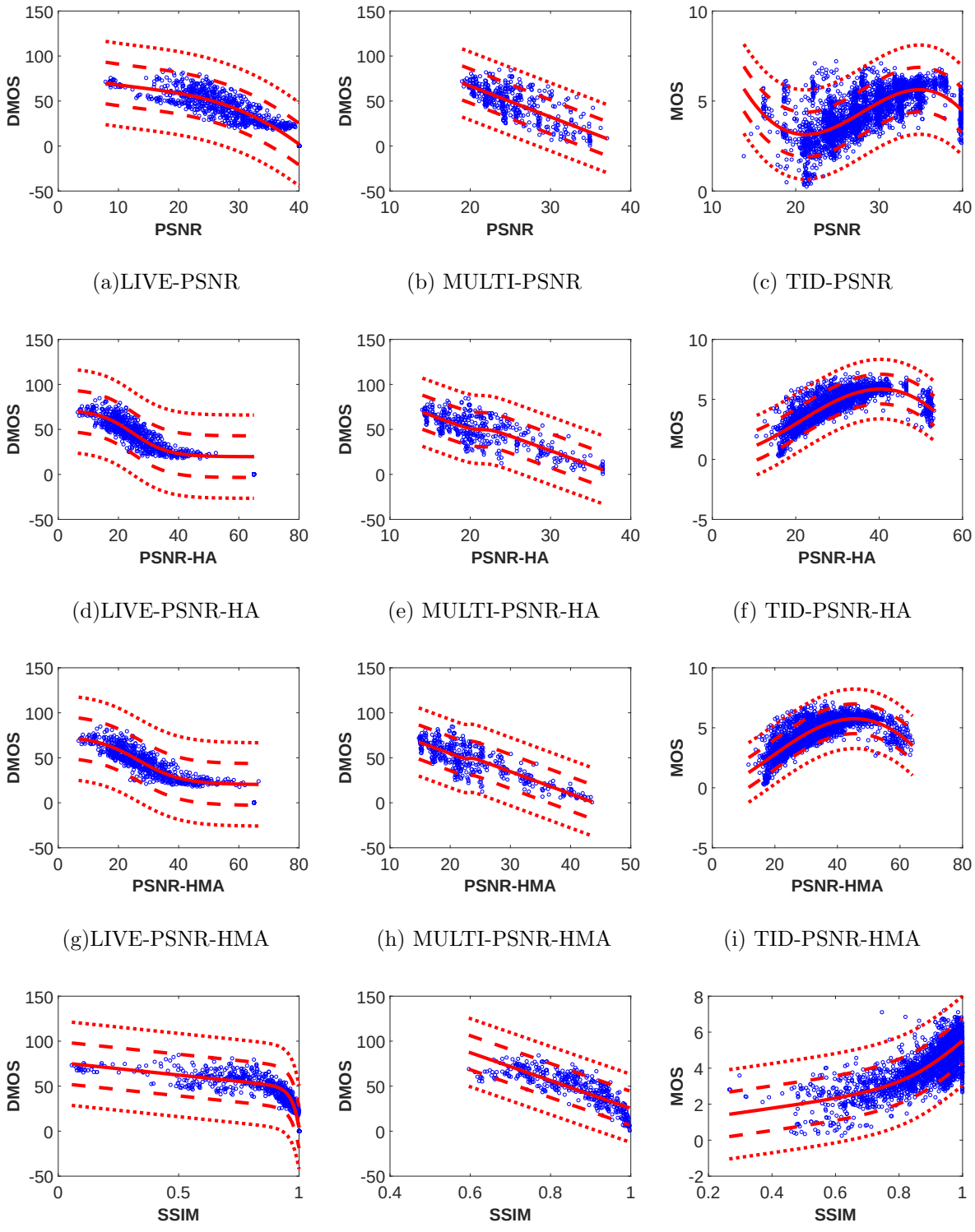
5.5.3 Scatter Plots and Histogram Differences

5.5.3.1 Scatter Plots

To analyze the distribution of subjective scores versus objective quality estimates, scatter plots of the quality estimators are given in Figs. 55, 56, 57, and 58. In these figures, x axis corresponds to the quality estimates and y axis corresponds to the mean opinion scores (MOS) or differential mean opinion scores (DMOS). We plot the non-linear mapping function that is learned by the regression formulation as a red curve in the scatter plots. Moreover, we also plot two curves that are one standard deviation away with dashed lines and two curves that are two standard deviations away with dotted lines. For an ideal quality estimator, scores should be located on a linear curve. Therefore, in practice, we target scattered points that follow a linear pattern with low deviation.

The scatter plots of PSNR, PSNR-HA, PSNR-HMA, and SSIM are shown in Fig. 55. The pattern of the scattered points in PSNR, PSNR-HA, and PSNR-HMA follow a monotonically decreasing behavior in the LIVE database, a linear behavior with high deviations in the MULTI database, and a parabolic behavior in the TID13 database. SSIM follows an almost monotonic behavior in all three databases. The scatter plots of MS-SSIM, CW-SSIM, IW-SSIM, and SR-SIM are shown in Fig. 56. MS-SSIM, IW-SSIM, and SR-SIM follow a monotonically decreasing behavior with a sharp decrease around high quality scores in the LIVE database, and a monotonic behavior in both the MULTI and the TID13 databases. CW-SSIM scores are mostly located around the ideal quality score 1.0 and regression formulation does not converge to a ideal monotonic mapping function between quality estimates and scores because of the outliers. The scatter plots of FSIM, FSIMc, BRISQUE, and BIQI are shown in Fig. 57. FSIMc is an extended version of FSIM and the main blocks in both quality estimators are same. Therefore, the scatter plots of FSIM and FSIMc

are also very similar. The points in the scatter plots follow a monotonically decreasing behavior in the LIVE database and a more linear behavior in the MULTI and the TID13 databases. BRISQUE and BIQI are no-reference methods and they are already regressed. Thus, the direction of the monotonic behavior is the opposite of other similarity-based methods. In the LIVE database, BIQI and BRISQUE follow a monotonically increasing behavior. However, we can observe a systematic error in the scatter plots of both of the methods, in which some of the images are assigned a zero DMOS whereas the objective quality scores corresponding to these images are varying. In the MULTI database, BIQI follows an almost monotonic behavior but BRISQUE follows a parabolic behavior. In the TID13 database, both BIQI and BRISQUE follow a monotonic behavior in most of the quality range whereas we can observe a slightly parabolic behavior when the values of objective quality scores are low. The scatter plots of BLIINDS2, `PerSIM`, `CSV`, and `UNIQUE` are shown in Fig. 58. BLIINDS2 follows a monotonic behavior in the LIVE database, a parabolic behavior in the MULTI database, and a linear behavior in the TID13 database. `PerSIM` follows an almost linear behavior in all the databases. `CSV` follows a monotonic behavior in the LIVE and the MULTI databases, and a linear behavior in the TID13 database. `UNIQUE` follows an almost linear behavior in the LIVE and the TID13 databases and a slightly monotonic behavior in the MULTI database. None of the introduced methods follow a parabolic behavior and even without any regression, the introduced methods are generally more linear compared to the existing methods. The majority of the existing non-regressed methods do not utilize the full quality range. On contrary, the introduced method `PerSIM` utilizes the full quality range in all the databases and `UNIQUE` utilizes the range in the LIVE and the TID13 databases.



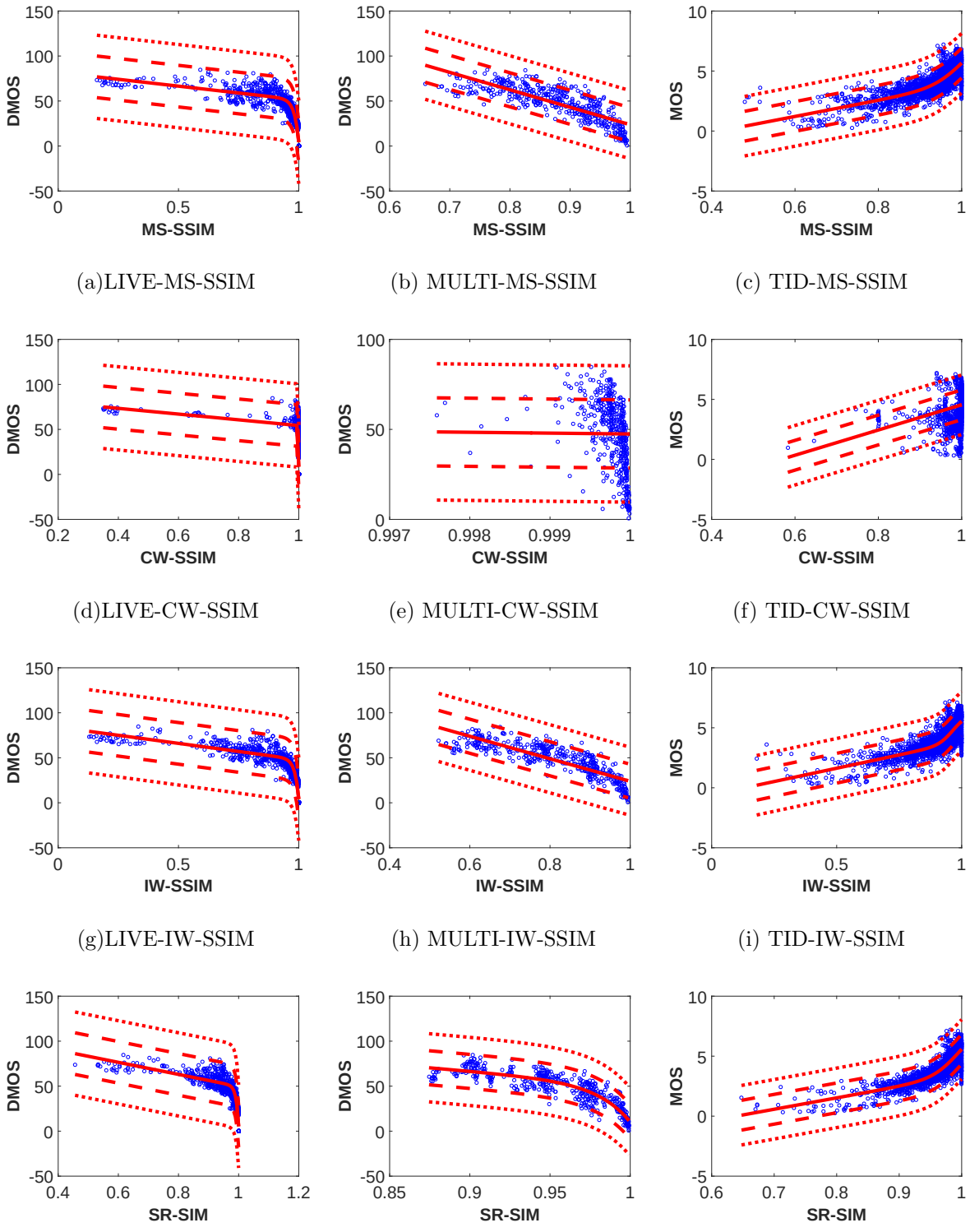
(a)LIVE-PSNR (b) MULTI-PSNR (c) TID-PSNR

(d)LIVE-PSNR-HA (e) MULTI-PSNR-HA (f) TID-PSNR-HA

(g)LIVE-PSNR-HMA (h) MULTI-PSNR-HMA (i) TID-PSNR-HMA

(j)LIVE-SSIM (k) MULTI-SSIM (l) TID-SSIM

Figure 55: Scatter plots of objective quality estimates PSNR, PSNR-HA, PSNR-HMA, and SSIM.



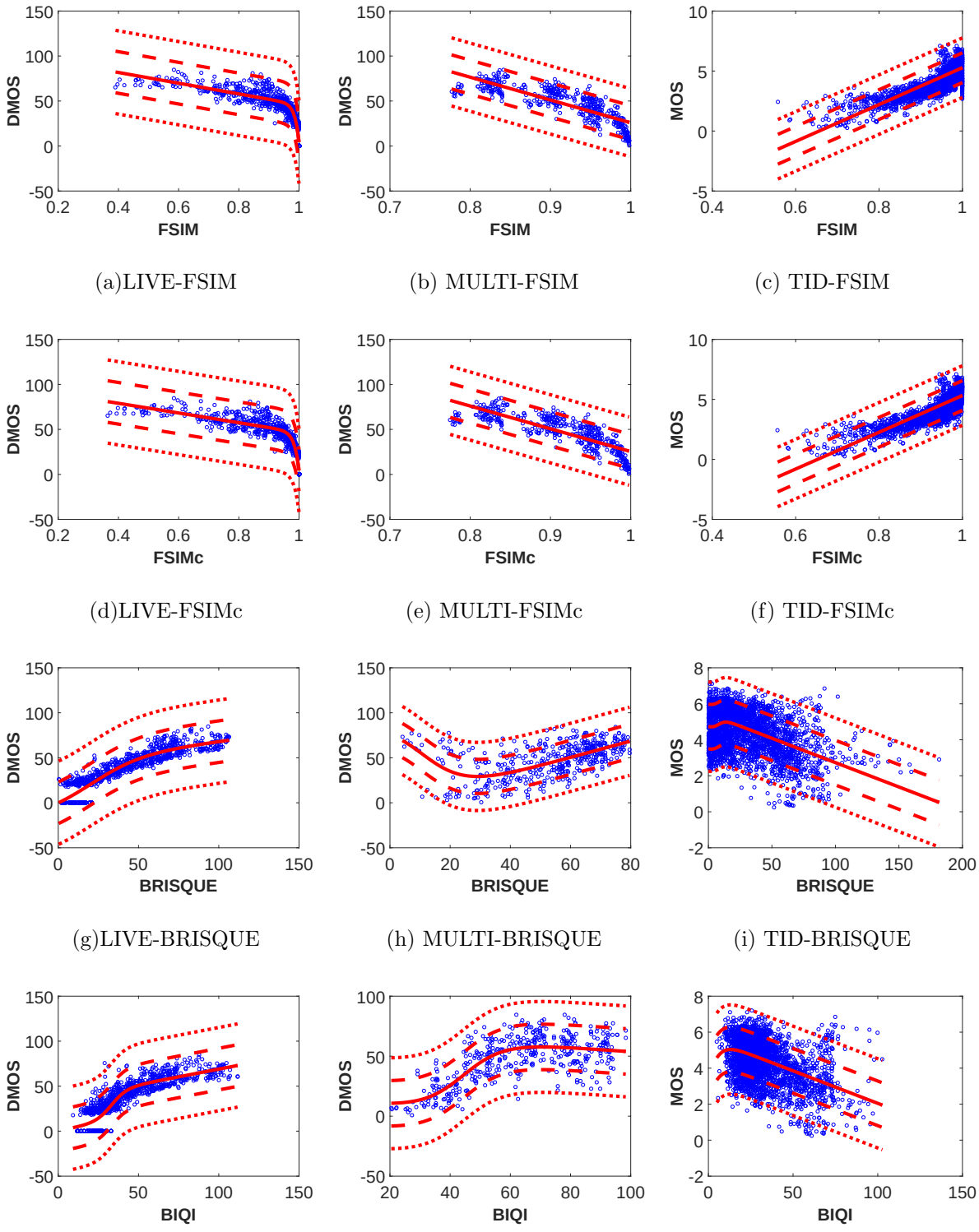
(a)LIVE-MS-SSIM (b) MULTI-MS-SSIM (c) TID-MS-SSIM

(d)LIVE-CW-SSIM (e) MULTI-CW-SSIM (f) TID-CW-SSIM

(g)LIVE-IW-SSIM (h) MULTI-IW-SSIM (i) TID-IW-SSIM

(j)LIVE-SR-SIM (k) MULTI-SR-SIM (l) TID-SR-SIM

Figure 56: Scatter plots of objective quality estimates MS-SSIM, CW-SSIM, IW-SSIM, and SR-SIM.



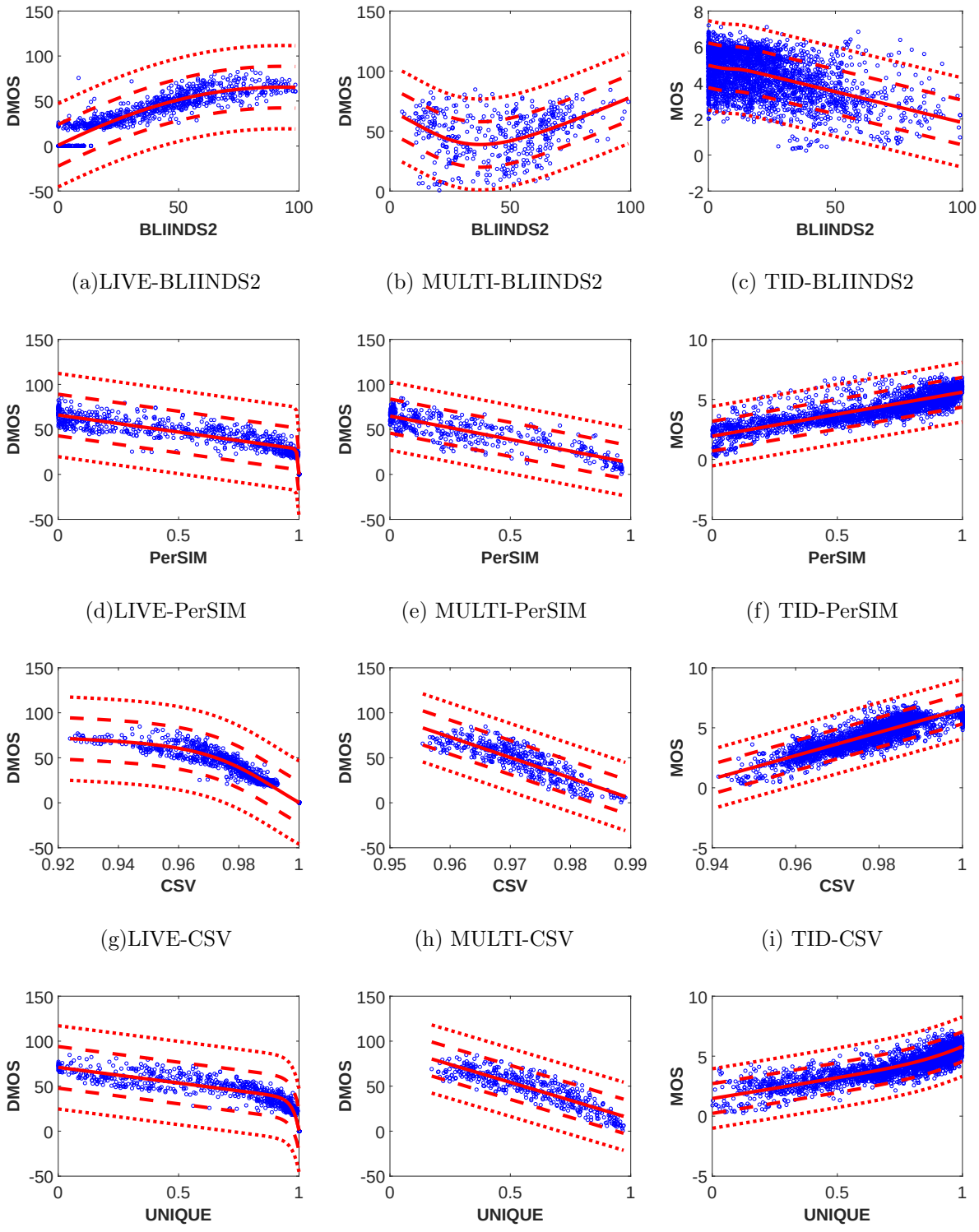
(a)LIVE-FSIM (b) MULTI-FSIM (c) TID-FSIM

(d)LIVE-FSIMc (e) MULTI-FSIMc (f) TID-FSIMc

(g)LIVE-BRISQUE (h) MULTI-BRISQUE (i) TID-BRISQUE

(j)LIVE-BIQI (k) MULTI-BIQI (l) TID-BIQI

Figure 57: Scatter plots of objective quality estimates FSIM, FSIMc, BRISQUE, and BIQI.



(a)LIVE-BLIINDS2 (b) MULTI-BLIINDS2 (c) TID-BLIINDS2

(d)LIVE-PerSIM (e) MULTI-PerSIM (f) TID-PerSIM

(g)LIVE-CSV (h) MULTI-CSV (i) TID-CSV

(j)LIVE-UNIQUE (k) MULTI-UNIQUE (l) TID-UNIQUE

Figure 58: Scatter plots of objective quality estimates BLIINDS2, PerSIM, CSV, and UNIQUE.

Table 18: Distributional difference between subjective scores and objective quality estimates.

	Existing Methods													Proposed Methods		
Methods	**PSNR**	**PSNR HA** [19]	**PSNR HMA** [19]	**SSIM** [130]	**MS SSIM** [14]	**CW SSIM** [45]	**IW SSIM** [16]	**SR SIM** [35]	**FSIM** [11]	**FSIM c** [11]	**BRIS QUE** [21]	**BIQI** [131]	**BLII NDS2** [132]	**Per SIM** [1]	**CSV** [2]	**UNI QUE** [4]
	LIVE Database															
EMD	0.260	0.237	**0.226**	0.283	0.319	0.491	0.297	0.325	0.276	0.272	0.495	0.502	0.502	0.353	**0.199**	**0.236**
KL	**0.246**	0.249	**0.205**	0.302	0.345	1.004	0.325	0.387	0.300	0.300	0.921	1.001	0.860	0.579	**0.189**	0.256
JS	0.057	0.060	**0.053**	0.065	0.075	0.172	0.072	0.085	0.066	0.066	0.182	0.188	0.154	0.109	**0.039**	**0.055**
HI	0.260	0.237	**0.226**	0.283	0.319	0.491	0.297	0.325	0.276	0.272	0.495	0.502	0.502	0.353	**0.199**	**0.236**
L2	0.070	0.070	**0.066**	0.072	0.079	0.249	0.076	0.084	0.072	0.071	0.227	0.230	0.196	0.112	**0.059**	**0.069**
	MULTI Database															
EMD	**0.324**	0.426	0.442	0.437	0.420	0.828	0.422	0.404	0.473	0.457	**0.324**	0.657	0.615	0.455	**0.264**	**0.308**
KL	**0.287**	0.431	0.458	0.481	0.461	1.691	0.470	0.420	0.536	0.516	0.306	1.412	0.844	0.653	**0.157**	**0.233**
JS	**0.062**	0.095	0.102	0.100	0.096	0.299	0.097	0.092	0.116	0.114	0.063	0.221	0.179	0.118	**0.038**	**0.053**
HI	**0.324**	0.426	0.442	0.437	0.420	0.828	0.422	0.404	0.473	0.457	**0.324**	0.657	0.615	0.455	**0.264**	**0.308**
L2	**0.091**	0.126	0.136	0.112	0.111	0.253	0.113	0.109	0.121	0.116	0.092	0.275	0.161	0.147	**0.070**	**0.080**
	TID13 Database															
EMD	0.550	**0.332**	**0.360**	0.468	0.446	0.946	0.500	0.507	0.699	0.689	0.679	0.559	0.605	0.496	**0.165**	0.404
KL	1.373	**0.821**	**0.927**	1.250	1.260	5.943	1.678	1.623	2.673	2.544	2.620	1.625	2.089	1.251	**0.173**	0.939
JS	0.214	**0.105**	**0.117**	0.165	0.157	0.566	0.196	0.193	0.319	0.309	0.322	0.218	0.255	0.173	**0.028**	0.130
HI	0.550	**0.332**	**0.360**	0.468	0.446	0.946	0.500	0.507	0.699	0.689	0.679	0.559	0.605	0.496	**0.165**	0.404
L2	0.150	**0.112**	0.124	0.143	0.144	0.834	0.180	0.175	0.258	0.239	0.278	0.168	0.231	0.138	**0.055**	**0.115**

5.5.3.2 Histogram Differences

We report the difference between the normalized histograms of subjective scores and the regressed quality estimates in Table 18. In each category, we highlight three best performing image quality assessment algorithms with a bold typeset. We highlight more than three methods when they lead to equivalent performances. Minimum difference leads to the best performing image quality estimator. PSNR is among the best performing methods in the MULTI database in all categories and in 1 category in the LIVE database. PSNR-HA is among the best performing methods in all the categories in the TID13 database. PSNR-HMA is among the best performing methods in all the categories in the LIVE database and in all the categories other than L2 in the TID13 database. BRISQUE is among the top performing methods in EMD and HI category in the MULTI database. The introduced method `CSV` is among the top performing methods in all the categories. `UNIQUE` is among the top performing methods in all the categories in the MULTI database, in all of the categories other than KL in the LIVE database, and in the L2 category in the TID13 database. Out of 15 categories, the existing methods that are among the top performing methods are PSNR in 6, PSNR-HA in 5, PSNR-HMA in 9, and BRISQUE in 2 categories. The introduced methods that are among the top performing quality estimators are `CSV` in all 15 categories and `UNIQUE` in 10 categories. The normalized histograms of the objective quality estimators are provided in the Appendix chapter as Figs. 61, 62, 63, and 64. We scale the range of the scores in the TID13 database to the same range with the LIVE and the MULTI databases in these figures.

5.5.4 Performance Evaluation of CSV Alternatives and UNIQUE's Extensions

5.5.4.1 Performance Evaluation of CSV Alternatives

In `CSV`, to measure the effect of interpolation strategy selection in image quality estimation performance, we use alternative interpolation strategies. Moreover, to

measure the effect of color difference formulation, we replace the color difference with an euclidean distance between color channels. CSV and its alternatives tested in this section can be summarized as follow:

- CSV: The original method introduced in [2], in which a bicubic method is used to interpolate color-based similarity maps to the original image resolution.
- $CSV_{bilinear}$: An alternative version of `CSV` [2], in which a bilinear method is used to interpolate color-based similarity maps to the original image resolution.
- $CSV_{nearest}$ An alternative version of `CSV` [2], in which a nearest neighbor method is used to interpolate color-based similarity maps to the original image resolution.
- $CSV_{euclidean}$ An alternative version of `CSV` [2], in which the color difference formulation is replaced with the Euclidean distance in the RGB color space.

Table 19: Performance of CSV and its alternatives.

Methods	**CSV**	$CSV_{bilinear}$	$CSV_{nearest}$	$CSV_{euclidean}$
	Pearson Correlation Coefficient			
LIVE	**0.967**	0.966	0.966	0.933
MULTI	**0.852**	0.850	0.850	0.828
	Spearman Correlation Coefficient			
LIVE	**0.959**	0.958	0.958	0.948
MULTI	**0.848**	0.846	0.846	0.814
TID13	0.845	**0.848**	0.847	0.797
	Kendall Correlation Coefficient			
LIVE	**0.834**	0.832	0.832	0.812
MULTI	**0.655**	0.651	0.652	0.619
TID13	0.654	**0.658**	0.656	0.605

In [2], we provide the results for the Spearman and the Kendall correlations in the LIVE, the MULTI, and the TID13 databases, and for the Pearson correlation in the LIVE and the MULTI databases. Therefore, in this section, we also provide the

results for these categories in Table 19. CSV outperforms $CSV_{euclidean}$ across all the databases and categories. We consider interpolation methods based on nearest neighbor, bilinear, bicubic, spline, and sinc. Bicubic interpolation is selected in the original version [2] because it is usually more accurate than nearest neighbor and bilinear, and more computationally efficient than spline and sinc. We avoid using spline- and sinc-based interpolation because of computational complexity; we provide the results for nearest neighbor and bilinear interpolation methods. In the LIVE and the MULTI databases, bicubic interpolation-based quality estimation slightly outperforms other methods in the overall databases in all correlation categories. However, in the TID13 database, bilinear slightly outperforms nearest, and nearest outperforms bicubic.

Table 20: Performance of UNIQUE and its extensions.

Methods	UNIQUE	MS-UNIQUE	DMS-UNIQUE
	Outlier Ratio		
MULTI	**0**	0.004	0.002
TID13	0.641	0.703	**0.639**
	Root Mean Square Error		
LIVE	6.76	**6.61**	6.63
MULTI	**9.25**	9.82	10.01
TID13	0.61	0.64	**0.57**
	Pearson Correlation Coefficient		
LIVE	0.956	**0.958**	**0.958**
MULTI	**0.872**	0.854	0.848
TID13	0.868	0.854	**0.883**
	Spearman Correlation Coefficient		
LIVE	**0.952**	0.949	0.948
MULTI	**0.866**	0.856	0.840
TID13	0.860	0.870	**0.880**

5.5.4.2 Performance Evaluation of UNIQUE's Extensions

The performance of UNIQUE and its extensions are summarized in Table 20. UNIQUE leads its extension in all the categories in the MULTI database. In the LIVE database,

the best performing methods are UNIQUE in the Spearman correlation and MS-UNIQUE in root mean square error. In terms of the Pearson correlation, MS-UNIQUE and DMS-UNIQUE lead UNIQUE. DMS-UNIQUE is the best performing method in all the categories in the TID13 database.

5.6 Performance Evaluation of Image Quality Assistance

We analyze the effect of BLeSS by focusing on relative performance changes percentage wise in terms of the Spearman correlation for FSIM, FSIM extended with chroma fidelity (FSIMc), and SR-SIM. Distortion category-based relative performance changes are provided in Table 21. Results are highlighted if there is an increase in the performance. We calculate performance changes in each database and provide the weighted average. In the case of communication distortions, we see a minor increase for all the quality maps. There are slight increases in the performance of FSIM and FSIMc in compression and blur category and relatively higher increases in local distortion category. In color distortion category, there is more than 100% increase for SR-SIM and FSIM, and there is around 10% increase in FSIMc. The increase in FSIMc is less compared to others since color-based similarity is already included in the quality estimator but BLeSS still enhances the performance. The overall performance changes in the case of BLeSS assistance is given in Table 22. The performance of FSIM and FSIMc increase for all databases whereas the performance of SR-SIM increases for the LIVE and the TID13 databases.

We perform statistical tests and analysis to verify that differences in terms of correlation coefficients are not solely random and they are statistically significant. In order to analyze the difference between correlation coefficients, we use statistical significance tests suggested in ITU-T Rec. P.1401. [117]. In Table 21 and Table 22, we report the statistical significance test results within parentheses next to the percentage change. In these test results, a 0 means that the change is not statistically

Table 21: Percentage performance changes for BLeSS-assisted image quality estimators over various distortion categories in terms of the Spearman correlation coefficient.

	SR-SIM	FSIM	FSIMc
Comp.	-0.29 (000)	**+0.13 (000)**	**+0.28 (000)**
Noise	-2.16 (001)	-1.31 (000)	-0.34 (000)
Comm.	**+0.07 (0-0)**	**+0.25 (0-0)**	**+0.24 (0-0)**
Blur	-0.39 (000)	**+0.20 (000)**	**+0.40 (000)**
Color	**+183 (–1)**	**+185 (–1)**	**+13.1 (–1)**
Global	-1.31 (–0)	-4.69 (–0)	-0.16 (–0)
Local	-1.85 (–0)	**+4.36 (–0)**	**+3.12 (–0)**

significant whereas a 1 corresponds to a statistically significant change. In Table 21, we provide the statistical significance for each distortion type and database. The first index corresponds to the LIVE database, the second index is for the MULTI database, and the third is for the TID13 database. If a specific database does not include a distortion type, there is a hyphen. The decrease in the performance of SR-SIM in the noise category of TID13 database is low. However, it is still statistically significant since the decrease is for 1,375 images. Moreover, the performance enhancement in color category is significant for all of the quality estimators. As summarized in Table 22, in full databases, the increases in FSIMc are not statistically significant whereas the increases in SR-SIM and FSIM are statistically significant in the TID13 database.

Table 22: Percentage performance changes for BLeSS-assisted image quality estimators over full databases in terms of the Spearman correlation coefficient.

	SR-SIM	FSIM	FSIMc
LIVE	**+0.13 (0)**	**+0.17 (0)**	**+0.06 (0)**
MULTI	-0.33 (0)	**+0.62 (0)**	**+0.79 (0)**
TID13	**+3.79 (1)**	**+4.77 (1)**	**+1.03 (0)**

5.7 Summary

As discussed in Section 2.5.1, pixel-wise fidelity does not highly correlate with subjective opinion, and structure and scale-space methods are well studied in the literature. Pooling strategy selection and color information are not commonly used in the literature and visual system-based studies require a better understanding of the perception process. Therefore, instead of focusing our attention on fidelity, structure, or scale-space, we focus on visual system and color to introduce two new image quality assessment methods `PerSIM` [1] and `CSV` [2], and a new image quality-assistance method `BLeSS` [3]. The characteristics of hand-crafted methods including PerSIM, CSV, and BLeSS are summarized in Table 23.

Table 23: Characteristics of hand-crafted methods including PerSIM, CSV, and BLeSS.

YEAR	Before 2000		2000	2003	2004	2006	2007		2008		2009		2011					2012			2014	2015		2016	
QUALITY ESTIMATORS	MSE-PSNR	PSNRc	NQM	MSSSIM	SSIM	PSNRHVS	PSNRHVSM	VSNR	VIF	C4	CWSSIM	Li-Wang	PSNRHA	PSNRHMA	FSIM	FSIMc	IWSSIM	CIEDE	BRISQUE	SR-SIM	CNN	REDLOG	PerSIM	CSV	BLeSS
Fidelity	■	■																					■		
Structure				■	■	■	■			■		■	■	■			■		■		■	■		■	
Scale Space			■	■		■	■	■	■		■	■	■	■	■	■	■		■	■		■	■		■
Visual System			■	■	■	■	■	■	■	■	■	■	■	■	■	■	■	■	■	■			■	■	■
Pooling											■						■								
Color		■								■						■		■					■	■	■

We introduce a multi-resolution image quality assessment method denoted as `PerSIM` [1], which is based on visual system characteristics and chroma similarity. Images are transformed from the RGB domain to the La*b* domain. The `L` channel is used to extract features through Laplacian of Gaussian operators, which partially formulate contrast sensitivity mechanisms of retinal ganglion cells. Color similarity is calculated over a* and b* channels. Feature map similarities are computed over

multiple resolutions to mimic the hierarchical nature of human visual system. Based on the validation in the LIVE, the MULTI, and the TID13 databases, `PerSIM` is generally among the top performing quality estimators.

We present an image quality estimator based on color, structure, and visual system characteristics denoted as `CSV` [2]. In contrast to the majority of existing methods, we quantify perceptual color degradations rather than absolute pixel-wise changes. We use CIEDE2000 color difference formulation to quantify the low-level color degradations and the Earth Mover's Distance between color name descriptors to measure significant color degradations. In addition to perceptual color difference, `CSV` also contains structural and perceptual differences. Structural feature maps are obtained by mean subtraction and divisive normalization, which mimic suppression mechanisms in cortical neurons. Perceptual feature maps are obtained by filtering with Laplacian of Gaussian operators, which formulate contrast sensitivity formulations of retinal ganglion cells. Primitive models of the contrast sensitivity and the suppression mechanisms along with the perceptual distance among color name descriptors are far away from being a comprehensive perceptual quality estimator. However, based on the validation in the LIVE, the MULTI, and the TID13 databases, `CSV` is still generally among the top performing quality estimators. Therefore, `CSV` articulates the importance of combining various perception mechanisms in a single quality estimator.

In the majority of the methods in the literature and the introduced methods `PerSIM` and `CSV`, color degradations are measured pixel-wise. However, the perception of the center is also affected by the surround in a visual system. Therefore, we introduce a biologically-inspired low-level spatiochromatic model-based similarity method (`BLeSS`) to assist full-reference image quality estimators that originally oversimplify color perception processes. More specifically, the spatiochromatic model is based on spatial frequency, spatial orientation, and surround contrast effects. The assistant similarity method is used to complement image quality estimators based on phase

congruency, gradient magnitude, and spectral residual. The effectiveness of BLeSS is validated using FSIM, FSIMc, and SR-SIM methods on the LIVE, the MULTI, and the TID13 databases. In terms of the Spearman correlation, BLeSS increases the quality assessment performance for feature similarity metrics in all the databases and for spectral residual-based metric in the LIVE and the TID13 databases. In the overall TID13 database, BleSS significantly enhances the performance of SR-SIM and FSIM. Moreover, significant changes in the color category lead to more than 100% enhancement for FSIM and SR-SIM.

In the introduced methods PerSIM [1], CSV [2], and BLeSS [3], visual system characteristics and color information are used in the design process. However, it is not possible to design a comprehensive quality estimator solely based on handcrafting because of the black box nature of a visual system. Therefore, in addition to visual system characteristics and color information, we also follow a data-driven approach. As explained in Section, 2.5.2, existing data-driven methods in the literature do not directly use pixel-wise fidelity whereas scale space, visual system, and pooling are used in some of the methods. Majority of the analyzed methods use structure and do not require a reference image as summarized in Table 24. To focus on the characteristics that are not well studied in the literature, in this thesis, we concentrate on data-driven approaches that use color and do not require handcrafting, distortion specific data or labels in the training. At first, we introduce the data-driven image quality estimator UNIQUE [4]. Then, we extend UNIQUE with multiple models and layers to obtain MS-UNIQUE [5] and DMS-UNIQUE.

In UNIQUE, we estimate perceived image quality using sparse representations obtained from generic image databases through an unsupervised learning approach. A color space transformation is used to perform operations in a perceptually correlated color space. A mean subtraction and a whitening operation are used to partially formulate suppression mechanisms in a visual system, which reduce spatial redundancy.

Table 24: Characteristics of data-driven methods including UNIQUE and its extensions. PS: We mark the method that is still in the submission phase with a '*'.

YEAR		2011		2012			2013			2014				2015					2016			
QUALITY ESTIMATORS		LBIQ	DIIVINE	CORNIA	BRISQUE	MLIQM	CB/SF	QAC	SPARQ	Tang	QAF	Kang	Q_{area} $Q_{exponent}$	IQA-CNN++	Li	S^2P^2	DLIQA	Gao	CNN-SVR	UNIQUE	MS-UNIQUE	DMS-UNIQUE*
Fidelity																						
Structure			■	■	■	■	■		■		■	■		■	■					■	■	■
Scale Space		■	■			■				■	■					■	■	■				
Visual system			■		■	■		■	■						■	■	■	■		■	■	■
Pooling		■		■				■			■	■			■		■		■			
Color						■														■	■	■
Do not require	Distortion specific data in the training								■				■							■	■	■
	Labels in the training							■	■				■							■	■	■
	Handcrafting			■			■					■		■					■	■	■	■
	Reference in testing	■	■	■	■		■	■		■	■	■	■	■	■	■	■	■	■			
Multiple layers/models without handcrafting												■		■	■				■		■	■

A linear decoder is used to obtain sparse representations, which mimic the visual representations in a visual system. And finally, a thresholding stage is used to formulate suppression mechanisms in a visual system. We train a linear decoder with 7 GB worth of data, which corresponds to $100,000$ 8×8 image patches randomly obtained from nearly $1,000$ images in the ImageNet 2013 database. A patch-wise training approach is preferred to maintain local information. Based on the validation in the LIVE, the MULTI, and the TID13 databases, `UNIQUE` is generally a top performing quality estimator in terms of accuracy, consistency, linearity, and monotonic behavior. The high performance of `UNIQUE` shows that unsupervised learning-based sparse representations, which do not require distortion specific data or subjective opinions in the training, can robustly estimate perceived quality. We extend `UNIQUE` by weighting learned filters with their sharpness and using multiple linear decoders with different number of hidden layers in `MS-UNIQUE`. We extend `MS-UNIQUE` by pooling the projected features that lead to maximum activations and adding an extra layer with

multiple linear decoders in DMS-UNIQUE. We also formulate how linear decoders can be stacked in the weight set generation phase to obtain deeper architectures. We show that UNIQUE and its extensions are generally among the top performing methods.

CHAPTER VI

SPATIAL POOLING AND METHOD BOOSTING

6.1 Spatial Pooling Strategy Selection

We describe visual representations that can capture local quality values in Chapter 3 and introduce quality estimators based on these representations in Chapter 5. As shown in Table 23 and Table 24, we directly use a single pooling mechanism in the introduced quality estimators without explicitly discussing the effect of the spatial pooling strategy. However, to understand the effect of pooling strategy selection in quality estimator design, we need to analyze the performance of quality estimators using different spatial pooling strategies. At first, we simplify the problem and introduce a toy example in Section 6.1.1. In this toy example, we pool entities in a 1-D array using commonly used pooling strategies. In Section 6.1.2, we extend the toy example using a quality map. Finally, we analyze the effect of spatial pooling strategy selection using image quality databases in Section 6.1.3.

6.1.1 Pooling in 1-D

We define a toy example vector as $A = [4, 3, 4, 3, 4, 3, 5, 1, 2, 2]$ and use the four pooling strategies described as

$$Mean(A) = \sum_{i=1}^{N} \frac{A_i}{N}, \tag{84}$$

$$Max(A) = A_m \; where \; A_m \geqslant A_i \; for \; \forall i, \tag{85}$$

$$Min(A) = A_m \; where \; A_m \leqslant A_i \; for \; \forall i, \tag{86}$$

$$Minkowski(A,p) = \sum_{i=1}^{N} \frac{A_i^p}{N}, \tag{87}$$

where A is an input vector, N is the number of entities in the vector, i is the index of the entities, m is the index of the entity that satisfies the required conditions, and p is the power of the entities in the input vector. The p values that are commonly used in the Minkowski pooling are 1/8, 1/4, 1/2, 2, 4, and 8. In addition to the pooling strategies defined in Eqs. (84) - (87), we also use median pooling, which outputs the entity in the middle of a sorted vector. If the number of elements in a vector are even, we take the average of the two entities in the middle to obtain the final value.

Table 25: Results of pooling using alternative strategies in 1-D.

Mean	**Min**	**Max**	**Median**	**Mink.(1/8)**	**Mink.(1/4)**	**Mink.(1/2)**	**Mink.(2)**	**Mink.(4)**	**Mink.(8)**
3.10	1.00	5.00	3.00	1.14	1.30	1.72	10.9	166.9	60,743

We can understand the effect of pooling strategy selection by analyzing the outputs of the pooling strategies summarized in Table 25. Average behavior of the entities are captured by mean and median pooling whereas extreme values are captured by min and max pooling. Minkowski pooling leads to significantly high values when p values are high and it converges to low values when p values are closer to zero. The pooling strategy selection significantly affects the pooled results in the toy example.

6.1.2 Spatial Pooling

To understand the effect of pooling strategy selection in image quality assessment, we extend pooling strategies described in Section 6.1.1 from 1-D to 2-D by performing pooling after vectorizing quality maps. Direct extension of these pooling methods from 1-D to 2-D is possible because they do not use any local information. As a toy example, we use the structural similarity (SSIM) map [14] corresponding to the images introduced in Section 3. SSIM is selected because it is a well studied and a commonly used method in the literature. In Fig. 59, we show the reference image

(a), the distorted image, (b) and the structural similarity map (c). We pool the SSIM map using alternative strategies and report the results in Table 26.

(a)Reference Image (b) Distorted Image (c) SSIM Map

Figure 59: Reference and distorted images with their structural similarity map.

Table 26: Results of spatial pooling using alternative strategies.

Mean	Min	Max	Median	Mink.(1/8)	Mink.(1/4)	Mink.(1/2)	Mink.(2)	Mink.(4)	Mink.(8)
0.94	0.57	0.99	0.97	0.95	0.93	0.92	0.87	0.81	0.73

Mean, median, and max pooling result in similar values because majority of the pixels in the quality map are close to 1.0. The range of Minkowski varies between 0.73 and 0.95 and min pooling leads to 0.57. The pooling strategy selection significantly affects the pooled results in quality map pooling. In the 1-D and the 2-D toy examples, we examine the changes in the pooled results and show that the final value can change significantly depending on the pooling strategy. However, it is not possible to obtain any conclusions about the performance of pooling strategies in terms of quality estimation without a validation set. To examine the effect of pooling strategy selection in image quality assessment, we use the validation methods described in Chapter 4.

6.1.3 The Effect of Spatial Pooling in Image Quality Estimation

We use the images and the subjective scores provided in the LIVE, the MULTI, and the TID13 databases to examine the relationship between objective and subjective

scores. Objective scores are functions of quality attributes and spatial pooling strategies. As quality attribute maps, we use the pixel-wise fidelity method squared error (SE), the structural-similarity method SSIM, and the perceptual-similarity method PerSIM. The squared error map corresponds to the pixel-wise square of the residual map, which is the difference of reference and degraded maps. Structural similarity is based on a full reference comparison, which includes luminance, contrast, and structure components that are calculated block-wise using the grayscale versions of a reference and a distorted image. Luminance is based on the mean and contrast is based on the standard deviation of the pixels in a local window. Structure is obtained by mean subtraction and division by standard deviation. The luminance, the contrast, and the structure maps of the reference and the degraded images are compared pixelwise to obtain the luminance, the contrast, and the structure similarity maps. Then, these maps are monotonically scaled according to their reliability and multiplicatively fused to obtain a single similarity map.

As a perceptual similarity method, we use `PerSIM`, which is described in Section 5.1. There are two main blocks in `PerSIM`. The first block is the pixel-wise Laplacian of Gaussian (LoG) similarity and the second block is the pixel-wise color similarity in chroma channels. These similarity maps are obtained from multiple resolutions and they are fused pixel-wise using a geometric mean. Individual maps are scaled according to the chroma sensitivity and pooled channel-wise using the min pooling operation. In the original implementation, squared error, SSIM, and `PerSIM` are pooled with a mean operation. However, in this chapter, we examine the effect of using alternative spatial pooling strategies. We use the spatial pooling strategies described in Section 6.1.2 over SE, SSIM, and `PerSIM` maps. Moreover, we also use percentile pooling [70], 5-Number summary [81], quality-weighted pooling, and information-weighted pooling [69] strategies, which are summarized in Table 27.

Table 27: Spatial pooling formulations of similarity maps. In case of dissimilarity maps, similarity maps (S) are replaced with dissimilarity maps (D) in the formulations.

Pooling Strategy	**Formulation**	Description
Mean	$\sum_{i=1}^{M \cdot N} \frac{S_i}{M \cdot N}$, (88)	i is the pixel index, M and N are the number of rows and columns.
Min	$S_m \; where \; S_m \leqslant S_i \; for \; \forall i$ (89)	Minimum operator.
Max	$S_m \; where \; S_m \geqslant S_i \; for \; \forall i$ (90)	Maximum operator.
Percentile [70]	$\begin{cases} S_i/c_1, & S_i < perc(P,S) \\ S_i, & \text{otherwise} \end{cases}$, (91)	i is the pixel index, $perc(\cdot)$ is the percentile function that returns the percentile of the values in the map, in which P is the target percentage in the interval $[0, 100]$. In case of the difference/distance maps, highly distorted entries are multiplied with the same constant.
5-Number Summary [81]	$\frac{mean + S1 + median + S3 + max}{5}$, (92)	S1 and S3 are equivalent to $perc(25, S)$ and $perc(75, S)$, respectively, and mean, median and max are the basic statistics calculated over the full resolution maps.
Minkowski [72]	$\sum_{i=1}^{M \cdot N} \frac{S_i^p}{M \cdot N}$, (93)	i is the pixel index, M and N are the number of rows and columns, respectively.
Quality Weighted[72]	$\frac{\sum_{i=1}^{M \cdot N} S_i \cdot W_i}{\sum_{i=1}^{M \cdot N} W_i}$, (94)	i is the pixel index, M and N are the number of rows and columns, the weight term is the p^{th} power of the pixel-wise similarity/dissimilarity value.
Information Weighted[69]	$log\left[\left(1 + \frac{(\sigma_i^R)^2}{c_2}\right)\left(1 + \frac{(\sigma_i^D)^2}{c_2}\right)\right]$, (95)	σ^R is the standard deviation map of the reference image, σ^D is the standard deviation map of the distorted image,and c_2 is a constant introduced to represent the channel noise.

The performance of image quality estimation using different spatial pooling strategies over various distortion types are given in Tables 28, 30, and 32 in which each table corresponds to a different quality attribute map. The abbreviations that correspond to the pooling strategies are summarized as follows:

- IW: Information-weighted
- 5-N: 5-Number summary
- M/M: Min for quality maps and max for distortion maps
- Per: Percentile
- MK: Minkowski
- QW: Quality-weighted

Image quality estimation performance in terms of the Spearman correlation are reported in three significant figures and the highest performing strategy is highlighted with a bold typeset. We do not provide more significant figures because minor differences do not lead to statistical significances as summarized in Tables 29, 31, 33. The format of the tables that provide statistical significance test results can be summarized as follows: We compare the pooling strategies in each row with all the pooling strategies sorted column-wise. Each pooling strategy comparison is performed in the LIVE, the MULTI, and the TID13 databases. In each database, we sum the entities column-wise to obtain the total number of statistically significant differences between the performance of spatial pooling strategies (DB Sum). We sum the total number of statistically significant differences in each pooling strategy type to obtain the total number of statistically significant performance comparisons in each database, which are reported in the last three columns. Finally, we sum the number of statistically

significant comparisons for each pooling strategy to obtain the total number of statistically significant comparisons in each database, which are reported as the last row in the last three columns.

6.1.3.1 Spatial Pooling of Squared-Error Maps

The performance of spatial pooling strategies in terms of the Spearman correlation coefficient using squared-error maps are summarized in Table 28. In terms of compression artifacts, quality-weighted , percentile, and Minkowski are the best performing pooling strategies. In the case of image noise, mean and Minkowski are the best performing pooling strategies. Percentile and Minkowski pooling are the best performing strategies in estimating the perceived quality of images degraded with communication-based distortions. Minkowski, percentile, and mean pooling are the best performers in blur-based degradation category. Color- and global-based degradations are best captured by Minkowski and local degradations are captured by percentile pooling. Overall, 5-number summary results in the highest Spearman correlation in the LIVE database, Minkowski in the MULTI database, and mean in the TID13 database.

In addition to identifying the pooling strategies that lead to the highest Spearman correlation, we also examine the magnitude of the differences between pooling strategy performance and their statistical significance. In the LIVE database, the maximum difference between the performance of pooling strategies is 0.06 in individual distortion types and 0.02 in the overall database. In the MULTI database, the maximum difference in individual categories is 0.1 and it is 0.08 in the overall database. In the TID13 database, the maximum difference in individual categories is 0.67 and it is 0.13 in the overall databases. The performance variations of the pooling strategies in the LIVE and the MULTI databases are smaller with respect to the variations in the TID13 database. The results of the statistical significance tests that compare pooling strategy performance using squared error maps are summarized in

Table 29. The performance of the pooling strategies are significantly different from each other in 10 comparisons in the LIVE database, in 2 comparisons in the MULTI database, and in 28 comparisons in the TID13 database out of 42 comparisons (7 methods × 6 compared methods) in each database.

Table 28: Performance of pooling strategies in terms of the Spearman correlation using squared error maps.

Distortion Type	Database	IW	5-N	M/M	Mean	Per	MK	QW
Comp.	Jp2k [LIVE]	0.954	0.973	0.973	0.953	0.965	0.973	**0.974**
	Jpeg [LIVE]	0.930	0.954	0.955	0.931	0.936	0.949	**0.956**
	Compression [TID13]	0.911	0.843	0.836	0.914	**0.914**	0.914	0.901
	Blur-Jpeg [MULTI]	0.639	0.731	0.730	0.662	0.670	**0.732**	0.718
Noise	Wn [LIVE]	0.991	0.989	0.989	0.991	0.990	**0.991**	0.991
	Blur-Noise [MULTI]	0.676	0.686	0.683	**0.708**	0.687	0.700	0.701
	Noise [TID13]	0.766	0.497	0.480	0.784	0.734	**0.801**	0.751
Comm.	FF [LIVE]	0.936	0.928	0.927	0.936	**0.942**	0.941	0.941
	Communication [TID13]	0.743	0.208	0.214	0.767	0.647	**0.871**	0.491
Blur	GBlur [LIVE]	0.865	0.921	0.921	0.873	0.895	**0.929**	0.914
	Blur [TID13]	0.890	0.820	0.802	0.895	**0.909**	0.908	0.880
	Blur-Jpeg [MULTI]	0.639	0.731	0.730	0.662	0.670	**0.732**	0.718
	Blur-Noise [MULTI]	0.676	0.686	0.683	**0.708**	0.687	0.700	0.701
Color	Color [TID13]	0.271	0.250	0.246	0.277	0.272	**0.749**	0.275
Global	Global [TID13]	0.513	0.567	0.499	0.502	0.522	**0.590**	0.576
Local	Local [TID13]	0.472	0.280	0.349	0.541	**0.568**	0.442	0.551
All	All [LIVE]	0.907	**0.924**	0.924	0.909	0.915	0.919	0.922
	All [Multi]	0.635	0.705	0.703	0.677	0.681	**0.714**	0.687
	All [TID13]	0.629	0.511	0.507	**0.639**	0.633	0.608	0.578

Table 29: Statistical significance results for spatial pooling strategies using squared error maps. PS: We use the following notations for the databases: L for LIVE, M for MULTI, and T for TID13.

	IW			**5-N**			**M/M**			**Mean**			**Per**			**MK**			**QW**			**Sum**		
	L	**M**	**T**	**L**	**M**	**T**	**L**	**M**	**T**	**L**	**M**	**T**	**L**	**M**	**T**	**L**	**M**	**T**	**L**	**M**	**T**	**L**	**M**	**T**
IW	0	0	0	1	0	1	1	0	1	0	0	0	0	0	0	0	1	0	1	0	1	**3**	**1**	**3**
5-N	1	0	1	0	0	0	0	0	0	1	0	1	0	0	1	0	0	1	0	0	1	**2**	**0**	**5**
M/M	1	0	1	0	0	0	0	0	0	1	0	1	0	0	1	0	0	1	0	0	1	**2**	**0**	**5**
Mean	0	0	0	1	0	1	1	0	1	0	0	0	0	0	0	0	0	1	0	0	1	**2**	**0**	**4**
Per	0	0	0	0	0	1	0	0	1	0	0	0	0	0	0	0	0	0	0	0	1	**0**	**0**	**3**
MK	0	1	0	0	0	1	0	0	1	0	0	1	0	0	0	0	0	0	0	0	0	**0**	**1**	**3**
QW	1	0	1	0	0	1	0	0	1	0	0	1	0	0	1	0	0	0	0	0	0	**1**	**0**	**5**
DB Sum	3	1	3	2	0	5	2	0	5	2	0	4	0	0	3	0	1	3	1	0	5	**10**	**2**	**28**

6.1.3.2 Spatial Pooling of Structural Similarity Maps

The performance of spatial pooling strategies in terms of the Spearman correlation coefficient using structural similarity (SSIM) maps are summarized in Table 30. 5-Number summary, Min/Max, and Minkowski are the best performing pooling strategies in the compression category. Min/Max and 5-Number summary are also best performing in the noise category. Perceived quality in the case of communication-based artifacts are best estimated by percentile and information-weighted pooling strategies. Perceived quality of blur-based degradations is best estimated by Min/Max and 5-Number summary. The best performing pooling strategy is Minkowski in the color and the local categories, and quality-weighted in the global category. Overall, Minkowski is the best performing pooling strategy in the LIVE and the TID13 databases whereas 5-Number summary is the best performing pooling strategy in the MULTI database.

The maximum difference between pooling strategy performance in the LIVE database is 0.03 in individual distortions and 0.01 in the overall database. In the MULTI database, the maximum difference in individual distortion groups is 0.03 and it is

0.03 in the overall database. In the TID13 database, the maximum difference in distortion groups is 0.81 and it is 0.17 in the overall database. The statistical significance results are summarized in Table 31. None of the differences are statistically significant in the LIVE and the MULTI databases. In the TID13 database, 26 comparisons out of 42 are statistically significant.

Table 30: Performance of pooling strategies in terms of the Spearman correlation using structural similarity maps.

Distortion Type	Database	IW	5-N	M/M	Mean	Per	MK	QW
Comp.	Jp2k [LIVE]	0.979	**0.981**	0.977	0.980	0.978	0.980	0.980
	Jpeg [LIVE]	0.962	0.962	0.960	0.962	0.961	**0.962**	0.962
	Compression [TID13]	0.931	**0.954**	0.902	0.935	0.927	0.938	0.933
	Blur-Jpeg [MULTI]	0.830	0.860	**0.866**	0.848	0.844	0.850	0.847
Noise	Wn [LIVE]	0.981	0.986	**0.989**	0.982	0.981	0.983	0.981
	Blur-Noise [MULTI]	0.862	**0.886**	0.873	0.876	0.870	0.879	0.875
	Noise [TID13]	0.837	**0.892**	0.856	0.847	0.823	0.852	0.845
Comm.	FF [LIVE]	**0.975**	0.968	0.944	0.974	0.972	0.974	0.973
	Communication [TID13]	0.889	0.841	0.085	0.894	**0.899**	0.895	0.894
Blur	GBlur [LIVE]	0.970	**0.975**	0.970	0.971	0.964	0.974	0.970
	Blur [TID13]	0.923	**0.927**	0.833	0.922	0.917	0.925	0.921
	Blur-Jpeg [MULTI]	0.830	0.860	**0.866**	0.848	0.844	0.850	0.847
	Blur-Noise [MULTI]	0.862	**0.886**	0.873	0.876	0.870	0.879	0.875
Color	Color [TID13]	0.231	0.244	0.239	0.234	0.241	**0.716**	0.234
Global	Global [TID13]	0.555	0.299	0.151	0.499	0.541	0.537	**0.569**
Local	Local [TID13]	0.275	0.390	0.366	0.288	0.181	**0.527**	0.415
All	All [LIVE]	0.949	0.946	0.941	0.949	0.948	**0.950**	0.949
	All [Multi]	0.845	**0.873**	0.871	0.860	0.854	0.863	0.859
	All [TID13]	0.747	0.727	0.594	0.741	0.660	**0.760**	0.731

Table 31: Statistical significance results for spatial pooling strategies using structural similarity maps. PS: We use the following notations for the databases: L for LIVE, M for MULTI, and T for TID13.

	IW			**5-N**			**M/M**			**Mean**			**Per**			**MK**			**QW**			**Sum**		
	L	**M**	**T**	**L**	**M**	**T**	**L**	**M**	**T**	**L**	**M**	**T**	**L**	**M**	**T**	**L**	**M**	**T**	**L**	**M**	**T**	**L**	**M**	**T**
IW	0	0	0	0	0	0	0	0	1	0	0	0	0	0	1	0	0	0	0	0	0	**0**	**0**	**2**
5-N	0	0	0	0	0	0	0	0	1	0	0	0	0	0	1	0	0	1	0	0	0	**0**	**0**	**3**
M/M	0	0	1	0	0	1	0	0	0	0	0	1	0	0	1	0	0	1	0	0	1	**0**	**0**	**6**
Mean	0	0	0	0	0	0	0	0	1	0	0	0	0	0	1	0	0	0	0	0	0	**0**	**0**	**2**
Per	0	0	1	0	0	1	0	0	1	0	0	1	0	0	0	0	0	1	0	0	1	**0**	**0**	**6**
MK	0	0	0	0	0	1	0	0	1	0	0	0	0	0	1	0	0	0	0	0	1	**0**	**0**	**4**
QW	0	0	0	0	0	0	0	0	1	0	0	0	0	0	1	0	0	1	0	0	0	**0**	**0**	**3**
DB Sum	0	0	2	0	0	3	0	0	6	0	0	2	0	0	6	0	0	4	0	0	3	**0**	**0**	**26**

6.1.3.3 Spatial Pooling of Perceptual Similarity Maps

The performance of spatial pooling strategies in terms of the Spearman correlation coefficient using PerSIM maps are summarized in Table 32. Minkowski and quality-weighted pooling are the best performing strategies in the compression and the image noise category. Minkowski and percentile pooling are the best performing strategies in estimating communication-based degradations. In the blur, the global, and the local categories, Minkowski is the best performing pooling strategy. Overall, Minkowski is the best performing pooling strategy in the LIVE and the MULTI databases, and mean pooling leads other strategies in the TID13 database.

The maximum difference between the performance of pooling strategies in individual distortion groups is 0.03 and it is 0.04 in the overall LIVE database. In the MULTI database, the maximum difference in individual distortion categories is 0.11 and it is 0.09 in the overall database. In the TID13 database, maximum difference in distortion groups is 0.42 and it is 0.17 in the full database. The statistical significance results are summarized in Table 33. The performance of the pooling strategies are significantly different from each other in 20 comparisons in the LIVE database, in 10

Table 32: Performance of pooling strategies in terms of the Spearman correlation using perceptual similarity maps.

Distortion Type	Database	IW	5-N	M/M	Mean	Per	MK	QW
Comp.	Jp2k [LIVE]	0.970	0.974	0.968	0.977	0.968	**0.978**	0.977
	Jpeg [LIVE]	0.951	0.954	0.952	0.958	0.952	0.959	**0.960**
	Compression [TID13]	0.961	0.963	0.961	0.964	0.960	**0.964**	0.963
	Blur-Jpeg [MULTI]	0.794	0.770	0.752	0.812	0.805	**0.817**	0.811
Noise	Wn [LIVE]	0.989	0.989	0.983	0.991	0.989	0.991	**0.992**
	Blur-Noise [MULTI]	0.795	0.775	0.713	0.818	0.813	**0.820**	0.817
	Noise [TID13]	0.921	0.852	0.836	0.924	0.900	**0.924**	0.924
Comm.	FF [LIVE]	0.944	0.935	0.912	0.945	0.941	**0.945**	0.945
	Communication [TID13]	0.858	0.790	0.703	0.865	**0.901**	0.886	0.888
Blur	GBlur [LIVE]	0.968	0.950	0.941	0.973	0.961	**0.973**	0.973
	Blur [TID13]	0.944	0.926	0.898	0.949	0.941	**0.951**	0.950
	Blur-Jpeg [MULTI]	0.794	0.770	0.752	0.812	0.805	**0.817**	0.811
	Blur-Noise [MULTI]	0.795	0.775	0.713	0.818	0.813	**0.820**	0.817
Color	Color [TID13]	0.739	0.455	0.418	0.762	**0.816**	0.777	0.790
Global	Global [TID13]	0.392	0.322	0.288	0.408	0.384	**0.409**	0.408
Local	Local [TID13]	0.427	0.128	0.456	0.470	0.189	**0.545**	0.431
All	All [LIVE]	0.945	0.926	0.916	0.950	0.946	**0.950**	0.950
	All [Multi]	0.797	0.776	0.737	0.818	0.811	**0.822**	0.817
	All [TID13]	0.844	0.732	0.688	**0.853**	0.754	0.853	0.853

comparisons in the MULTI database, and in 28 comparisons in the TID13 database out of 42 comparisons in each database.

6.1.3.4 Comparison of Spatial Pooling Strategies

We select the best performing spatial pooling strategy using a specific quality or distortion map in each distortion type and report the results in Table 34. Even with the best performing pooling strategy, squared error is the highest performing quality estimator only in the global and the local distortion categories. SSIM is the best performing quality estimator in most of the categories in the LIVE and the MULTI

Table 33: Statistical significance results for spatial pooling strategies using perceptual similarity maps. PS: We use the following notations for the databases: L for LIVE, M for MULTI, and T for TID13.

	IW			**5-N**			**M/M**			**Mean**			**Per**			**MK**			**QW**			**Sum**		
	L	**M**	**T**	**L**	**M**	**T**	**L**	**M**	**T**	**L**	**M**	**T**	**L**	**M**	**T**	**L**	**M**	**T**	**L**	**M**	**T**	**L**	**M**	**T**
IW	0	0	0	1	0	1	1	1	1	0	0	0	0	0	1	0	0	0	0	0	0	**2**	**1**	**3**
5-N	1	0	1	0	0	0	0	0	1	1	0	1	1	0	0	1	0	1	1	0	1	**5**	**0**	**5**
M/M	1	1	1	0	0	1	0	0	0	1	1	1	1	1	1	1	1	1	1	1	1	**5**	**5**	**6**
Mean	0	0	0	1	0	1	1	1	1	0	0	0	0	0	1	0	0	0	0	0	0	**2**	**1**	**3**
Per	0	0	1	1	0	0	1	1	1	0	0	1	0	0	0	0	0	1	0	0	1	**2**	**1**	**5**
MK	0	0	0	1	0	1	1	1	1	0	0	0	0	0	1	0	0	0	0	0	0	**2**	**1**	**3**
QW	0	0	0	1	0	1	1	1	1	0	0	0	0	0	1	0	0	0	0	0	0	**2**	**1**	**3**
DB Sum	2	1	3	5	0	5	5	5	6	2	1	3	2	1	5	2	1	3	2	1	3	**20**	**10**	**28**

Table 34: Performance of best pooling strategies for different quality attributes.

Distortion Type	**Database**	**SE**		**SSIM**		**PerSIM**	
Compression	**Jp2k [LIVE]**	QW	0.974	**5-N**	**0.981**	MK	0.978
	Jpeg [LIVE]	QW	0.956	**MK**	**0.962**	QW	0.960
	Compression [TID13]	Per	0.914	5-N	0.954	**MK**	**0.964**
	Blur-Jpeg [MULTI]	QW	0.732	**M/M**	**0.866**	MK	0.817
Noise	**Wn [LIVE]**	MK	0.991	M/M	0.989	**QW**	**0.992**
	Blur-Noise [MULTI]	Mean	0.708	**5-N**	**0.886**	MK	0.820
	Noise [TID13]	MK	0.801	5-N	0.892	**MK**	**0.924**
Communication	**FF [LIVE]**	Per	0.942	**IW**	**0.975**	MK	0.945
	Communication [TID13]	MK	0.871	Per	0.899	**Per**	**0.901**
Blur	**GBlur [LIVE]**	MK	0.929	**5-N**	**0.975**	MK	0.973
	Blur [TID13]	Per	0.909	5-N	0.927	**MK**	**0.951**
	Blur-Jpeg [MULTI]	MK	0.732	**M/M**	**0.866**	MK	0.817
	Blur-Noise [MULTI]	Mean	0.708	**5-N**	**0.886**	MK	0.820
Color	**Color [TID13]**	MK	0.749	MK	0.716	**Per**	**0.816**
Global	**Global [TID13]**	**MK**	**0.590**	QW	0.569	MK	0.409
Local	**Local [TID13]**	**Per**	**0.568**	MK	0.527	MK	0.545
All	**All [LIVE]**	5-N	0.924	MK	0.950	**MK**	**0.950**
	All [MULTI]	MK	0.714	**5-N**	**0.873**	MK	0.822
	All [TID13]	Mean	0.639	MK	0.760	**Mean**	**0.853**

Table 35: Statistical significance results for spatial pooling strategies using squared error, structural similarity, and perceptual similarity maps.

	Overall								
	LIVE			**MULTI**			**TID13**		
IW	3	0	2	1	0	1	3	2	3
5-N	2	0	5	0	0	0	5	3	5
M/M	2	0	5	0	0	5	5	6	6
Mean	2	0	2	0	0	1	4	2	3
Per	0	0	2	0	0	1	3	6	5
MK	0	0	2	1	0	1	3	4	3
QW	1	0	2	0	0	1	5	3	3
Col. Sum	10	0	20	2	0	10	28	26	28
DB Sum	30			12			82		

databases, and `PerSIM` is the best performing in most of the distortion categories in the TID13 database other than the ones leaded by SE.

The statistical significance of the difference between the Spearman correlation coefficients of the pooling strategies are summarized in Table 35. The sum of the individual columns (Col. Sum) corresponds to the total statistical difference in a specific database using a specific attribute. For each quality attribute, TID13 leads to the highest statistical significance total. When the statistical significance totals are summed up for each database (DB Sum), changes in the performance lead to 30 statistically significant differences in the LIVE database, 12 in the MULTI database, and 82 in the TID13 database out of 126 comparisons (7 methods $\times$ 6 compared methods $\times$ 3 quality attributes) in each database.

When we select the spatial pooling strategies that lead to best performing results, SSIM generally outperforms other attributes in the LIVE and the MULTI database, and `PerSIM` generally outperforms others in the TID13 database. Therefore, even spatial pooling strategy selection can lead to performance changes, quality attribute selection is still the key in determining the performance, especially when generic

pooling strategies are used over pre-designed quality attributes. We also observe that as the number of degradation types and images increase in a validation set, it becomes easier to differentiate the performance of spatial pooling strategies. MULTI database includes less number of images and degradation types than other databases and TID13 includes the most. Meanwhile, TID13 leads to the highest number of statistically significant comparisons and MULTI leads to the lowest.

6.1.4 Summary

Existing studies in spatial pooling strategy selection for image quality assessment generally use a single type of quality attribute while providing a comprehensive analysis. The studies utilizing multiple quality attributes perform a less comprehensive analysis that does not include statistical significance tests. However, in [6], we perform a comprehensive analysis of multiple quality attributes over different degradation types using various pooling strategies as summarized in Table 36.

In this thesis and in [6], we analyze the effect of spatial pooling strategy selection in image quality assessment. We compare the performance of spatial pooling strategies including information-weighted pooling, 5-Number summary, min/max pooling, percentile pooling, Minkowski, and quality-weighted pooling. Images in the test set are degraded with compression, image noise, communication error, blur, color artifacts, global artifacts, and local artifacts. As quality attributes, we use squared error, structural similarity [130], and perceptual similarity [1]. Based on the comparison of spatial pooling strategies, we conclude that pooling strategies influence the performance of an estimator. However, quality or distortion map selection is still more dominant in the performance of perceived quality estimation. It is easier to analyze the effect of pooling strategies or quality attributes as the number of distortion types increase in the validation set.

Table 36: Characteristics of spatial pooling strategy selection studies including the thesis work

YEAR		2007	2009	2012	2014	2014	2015	2016
EXISTING STUDIES		Wang	Moorthy	Gong	Zewdie	Bruni	Temel	Li
Spatial Pooling Strategies	Mean	■	■	■	■	■	■	■
	Min/Max						■	
	Minkowski	■		■	■		■	
	Quality/Distortion Weighted	■		■	■		■	
	Percentile Pooling		■		■		■	
	Information-weighted	■		■		■	■	
	Others		Fixation	Saliency(IG, NB)	5-N	Fixation, L-C	WPP, 5-N	STD, CoV, SP
Distortion Categories	Compression	■	■		■	■	■	■
	Image Noise	■	■		■	■	■	■
	Communication	■	■		■	■	■	■
	Blur	■	■		■	■	■	■
	Color				■		■	
	Global				■	■	■	■
	Local				■	■	■	■
	Others			Rendering Methods				
Quality Attribute Types	Squared Error	■					■	
	Structural Similarity	■	■	■	■	■	■	■
	Others			ΔLC, ABF, S-CIELAB, S-DEE, WLF			PerSIM	GMS, FSIM
Statistical Significance Tests					■		■	

6.2 Boosting-based Image Quality Estimators

Existing image quality estimators differ from each other in various ways. However, all these methods fundamentally map pixels to subjective scores. Moreover, even some of the methods are less perceptually correlated than others, they can still contain additional information that is not provided by better performing methods. Therefore, multiple methods can be fused to boost the overall performance. In this thesis and in [7], we analyze the effect of boosting in image quality assessment through multi-method fusion. Existing multi-method studies focus on proposing a single quality estimator. On the contrary, we investigate the generalizability of multi-method fusion as a framework. In addition to support vector machines that are commonly used in the multi-method fusion, we propose using neural networks in boosting. To span different types of image quality assessment algorithms, we use quality estimators based on fidelity, perceptually-extended fidelity, structural similarity, spectral similarity, color,

and learning.

6.2.1 Image Quality Estimators Utilized in Boosting

6.2.1.1 Fidelity-based

Fidelity attributes quantify the changes in a degraded image with respect to a reference image and they are commonly preferred in image and video coding standards for rate-distortion optimization because of low computational complexity and ease of implementation. The intuitive method to measure the fidelity of an image is to directly compare it with its distortion-free image, if available. Mean square error (MSE) is a commonly used pixel-wise fidelity method, which is calculated by obtaining the difference between images, taking the square root of the difference, and calculating the mean value. MSE is scaled by the range of an image and mapped with a logarithmic function to obtain the peak signal-to-noise ratio (PSNR), which is one of the quality estimators used in boosting operations.

6.2.1.2 Perceptually Extended Fidelity-based

Image quality metrics use the characteristics known about the visual system to make the perceptual quality assessment more accurate. The authors in [17] extend PSNR by removing any mean shift, stretching contrast block-wise, and quantizing DCT coefficients with the compression table proposed by JPEG. These extensions are performed to make PSNR compatible with the human visual system and the extended metric is named as PSNR-HVS. Reduction by value of contrast masking is also added to the metric and the modified version is named as PSNR-HVS-M [18]. These metrics are further extended by adding contrast change and mean shifting sensitivity (PSNR-HA, PSNR-HMA) as explained in [19], both of which are used in boosting operations.

6.2.1.3 Structural Similarity-based

Structural similarity is commonly obtained by quantifying the similarity between mean subtracted and divisive normalized images. The authors in [14] propose a full reference metric (SSIM) based on the comparison between a reference and a distorted image in terms of luminance, contrast, and structure in the spatial domain. These structure-based methods are also extended to multi-scale (MS-SSIM) [14], complex domain (CW-SSIM) [15], and information-weighted (IW-SSIM) [16] versions. All of these structural similarity methods are used in boosting operations. Moreover, we also use spectral similarity in boosting [35].

6.2.1.4 Color-based

The human visual system (HVS) is more sensitive to changes in intensity compared to color [27]. Although color may not be as informative as intensity, it can still contain additional information. An intuitive way to use color information in image quality assessment is pixel-wise fidelity. FSIMc [11] and `PerSIM` [1] introduce color information by computing pixel-wise fidelity over chroma channels in the La*b* color space. In addition to the color-based similarity, FSIMc computes similarity based on phase congruency and gradient magnitude, and `PerSIM` computes similarity based on band-pass features that are obtained from the contrast sensitivity formulation of the retinal ganglion cells. FSIMc are `PerSIM` are used in boosting operations.

6.2.1.5 Learning-based

It is not possible to handcraft a comprehensive quality estimator that covers all the aspects of visual system. Therefore, data-driven approaches can be used to design quality estimators. The majority of the data-driven approaches require distortion-specific images or subjective scores in the training, which can bias the performance of boosting methods. Therefore, we use the data-driven quality estimator UNIQUE

[4] in boosting, which is trained with solely generic images in an unsupervised fashion. Images are preprocessed with a mean subtraction stage, a whitening operation, and color space transformations to obtain more descriptive representations in terms of structure and color. These representations are fed to a linear decoder to obtain sparse representations. An objective score is obtained by comparing the sparse representations in terms of monotonic behavior.

6.2.2 Boosting Methods

Rather than using specifically tuned deep networks or complicated architectures, we analyze the effect of boosting through two off-the-shelf methods. We use a generic neural network and a support vector machine. The only parameter that we adjust in the neural network architecture is the number of neurons in a single hidden layer, which is set to the total number of quality estimators used in the experiments. By default, we use mean square error as the cost function and Levenberg-Marquardt as the training function, which does not necessarily guarantee a global minimum. The default configuration in a support vector machine includes a sequential minimal optimization (SMO) as the solver and a linear kernel.

6.2.3 Performance Evaluation

6.2.3.1 Data Partitioning and Number of Experiments

In the experiments, performance of the quality estimators are measured with k-fold cross validation, in which k is set to 5. At each iteration, 20% of total images in each database are selected as the test set. In Section 6.2.3.3, we test the performance of methods boosted with a neural network and a support vector machine. Each method is trained and tested 100 times. The test set in each iteration is also used to measure the performance of existing quality estimators. Since there are 11 different quality estimators, 2 boosting methods, and 100 runs, we report the average performance of existing quality estimators for $2,200$ runs in Section 6.2.3.2.

Table 37: Performance of existing image quality estimators using 5-fold validation for 2,200 runs.

Methods	PSNR	PSNR HA	PSNR HMA	SSIM	MS SSIM	CW SSIM	IW SSIM	SR SIM	FSIMc	PerSIM	UNIQUE
	Root Mean Square Error										
LIVE	8.60	6.92	**6.57**	7.51	7.42	11.3	7.09	7.53	7.19	6.79	6.75
MULTI	12.7	11.2	10.7	11.0	11.2	18.8	10.0	**8.68**	10.7	9.89	9.24
TID13	0.87	0.65	0.69	0.76	0.69	1.20	0.68	0.61	0.68	0.64	**0.61**
	Pearson Correlation Coefficicent										
LIVE	0.927	0.953	**0.958**	0.945	0.947	0.871	0.951	0.945	0.950	0.955	0.956
MULTI	0.737	0.799	0.819	0.813	0.803	0.406	0.846	**0.887**	0.820	0.850	0.871
TID13	0.705	0.850	0.827	0.788	0.830	0.228	0.831	0.866	0.832	0.854	**0.868**
	Spearman Correlation Coefficient										
LIVE	0.907	0.936	0.942	0.947	0.949	0.900	**0.959**	0.954	0.958	0.949	0.950
MULTI	0.672	0.709	0.738	0.855	0.831	0.626	**0.878**	0.860	0.860	0.812	0.861
TID13	0.700	0.846	0.816	0.740	0.784	0.562	0.776	0.806	0.850	0.852	**0.859**

6.2.3.2 Part 1

We report the performance of existing quality estimators in Table 37. In terms of root mean square error and the Pearson correlation, the best performing methods are PSNR-HMA in the LIVE database and SR-SIM in the MULTI database. In terms of the Spearman correlation, IW-SSIM is the best performing method in the LIVE and the MULTI databases. UNIQUE is the best performing quality estimator in terms of all the metrics in the TID13 database.

Neural network-based regression results are given in Table 38. Neural networks trained with fidelity-, perceptually-extended fidelity-, and perceptual similarity-based methods enhance the performances in some categories and degrade in others with minor changes. In terms of root mean square error and the Pearson correlation, neural networks lead to significant or minor enhancements for structural, spectral, unsupervised learning-based, and feature-based similarity methods.

In terms of the Spearman correlation, regressing existing quality estimators with neural networks leads to minor performance changes for most of the methods and

Table 38: Performance of image quality estimators with neural network-based regression using 5-fold validation for 100 runs.

Methods	PSNR	PSNR HA	PSNR HMA	SSIM	MS SSIM	CW SSIM	IW SSIM	SR SIM	FSIMc	PerSIM	UNIQUE
	Root Mean Square Error										
LIVE	8.21	6.93	6.61	5.98	5.91	8.81	**5.49**	5.80	5.56	6.35	6.09
MULTI	13.3	11.4	10.8	8.68	9.56	14.6	**7.91**	8.29	8.12	9.76	8.80
TID13	0.86	0.64	0.68	0.72	0.64	1.70	0.68	**0.59**	0.60	0.62	0.61
	Pearson Correlation Coefficicent										
LIVE	0.934	0.954	0.957	0.966	0.967	0.923	**0.971**	0.967	0.970	0.961	0.964
MULTI	0.710	0.793	0.821	0.890	0.866	0.646	**0.907**	0.900	0.903	0.855	0.886
TID13	0.722	0.852	0.830	0.814	0.852	0.442	0.836	**0.879**	0.872	0.864	0.870
	Spearman Correlation Coefficient										
LIVE	0.904	0.937	0.941	0.947	0.950	0.894	**0.958**	0.954	0.957	0.948	0.950
MULTI	0.660	0.701	0.738	0.858	0.827	0.613	**0.883**	0.874	0.872	0.797	0.861
TID13	0.706	0.845	0.813	0.794	0.834	0.558	0.807	0.843	0.850	0.852	**0.860**

Table 39: Performance of image quality estimators with support vector machine-based regression using 5-fold validation for 100 runs.

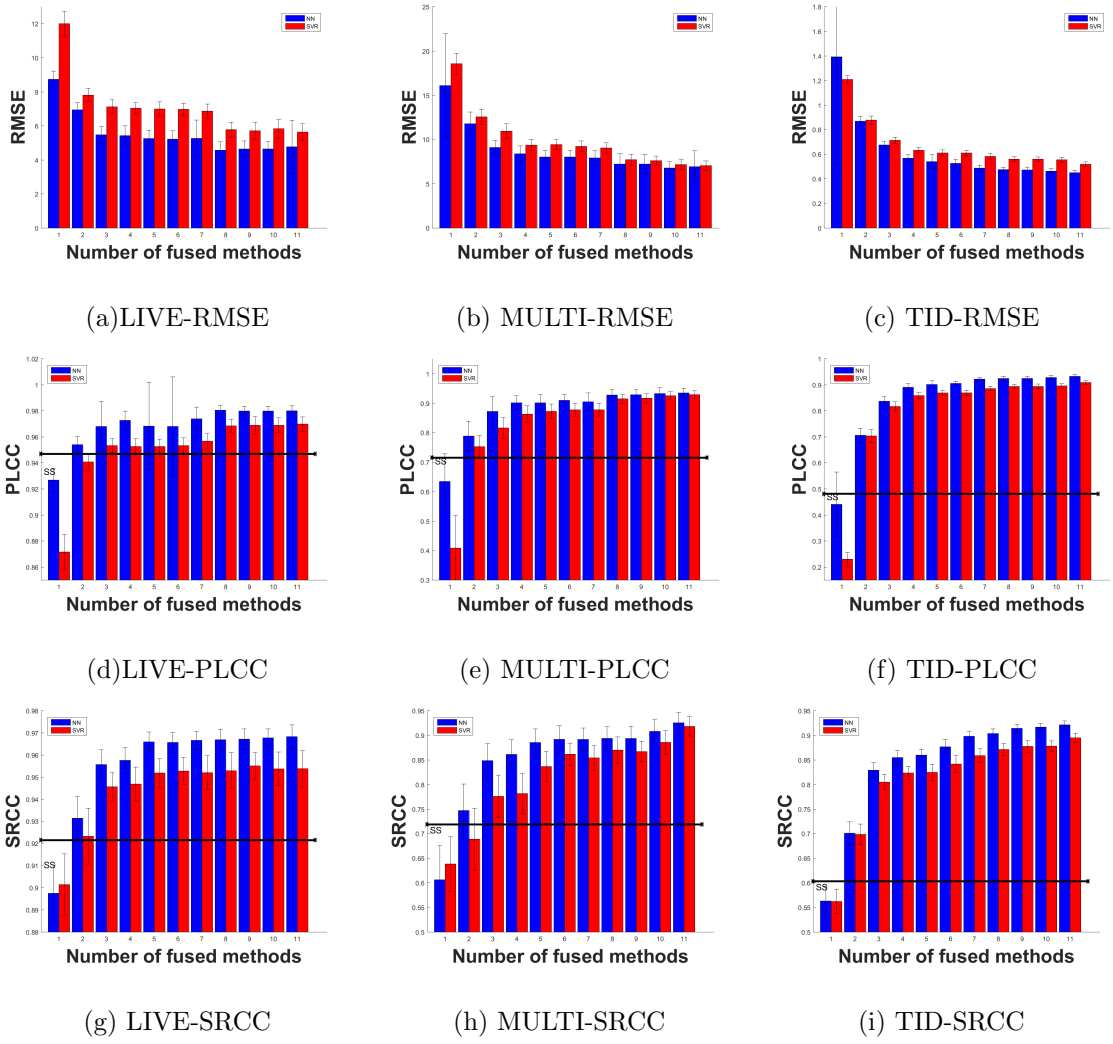
Methods	PSNR	PSNR HA	PSNR HMA	SSIM	MS SSIM	CW SSIM	IW SSIM	SR SIM	FSIMc	PerSIM	UNIQUE
	Root Mean Square Error										
LIVE	8.58	7.00	**6.66**	7.58	7.51	11.90	7.10	7.76	7.27	6.89	6.79
MULTI	13.1	11.3	10.7	11.0	11.3	18.5	10.0	**8.75**	10.9	10.0	9.23
TID13	0.88	0.66	0.70	0.77	0.69	1.21	0.69	0.62	0.69	0.64	**0.61**
	Pearson Correlation Coefficient										
LIVE	0.928	0.953	**0.958**	0.945	0.947	0.871	0.952	0.945	0.951	0.955	0.956
MULTI	0.720	0.798	0.820	0.814	0.798	0.399	0.851	**0.887**	0.816	0.847	0.874
TID13	0.706	0.848	0.826	0.787	0.828	0.226	0.829	0.866	0.833	0.855	**0.869**
	Spearman Correlation Coefficient										
LIVE	0.908	0.935	0.942	0.947	0.950	0.901	**0.960**	0.953	0.958	0.948	0.951
MULTI	0.652	0.714	0.738	0.855	0.828	0.616	**0.882**	0.860	0.855	0.813	0.864
TID13	0.701	0.843	0.814	0.736	0.785	0.558	0.773	0.807	0.849	0.853	**0.859**

databases. However, we also observe major performance changes for some of the quality estimators in the TID13 database. After the neural network-based regression, IW-SSIM becomes the best performing quality estimator in terms of root mean square error and the Pearson correlation in the LIVE and the MULTI databases. In the TID13 database, SR-SIM becomes the best performing quality estimator in terms of root mean square error and the Pearson correlation after the neural network-based regression. We also perform support vector machine-based regression and the results

are given in Table 39. The types of the quality estimators that lead to the best performance with and without support vector regression are same. In Table 40, we report the best performance values of existing and regressed methods. Moreover, we also report the performances of neural network- and support vector machine-based boosting. Existing methods regressed with neural networks perform better than existing methods in all the categories other than Spearman in the LIVE database, and the performances of existing methods regressed with support vector machines are similar to existing methods. Support vector machine-based boosting performs better than existing and regressed existing methods in the MULTI and the TID13 databases, whereas in the LIVE database, it is better in some categories and worse in others. Neural network-based boosting leads to the best performance in all the categories.

Table 40: Performance of existing, regressed, and boosted image quality estimators. In the case of existing and regressed methods, we report the best performing quality estimators.

	Existing Method	**NN Regression**	**SVM Regression**	**NN Boosting**	**SVM Boosting**
	Root Mean Square Error				
LIVE	6.57	5.49	6.66	4.54	5.62
MULTI	8.68	7.91	8.75	6.73	7.07
TID13	0.61	0.59	0.61	0.45	0.51
	Pearson Correlation Coefficicent				
LIVE	0.958	0.971	0.958	0.980	0.970
MULTI	0.887	0.907	0.887	0.934	0.926
TID13	0.868	0.879	0.869	0.931	0.909
	Spearman Correlation Coefficient				
LIVE	0.959	0.958	0.960	0.969	0.956
MULTI	0.878	0.883	0.882	0.918	0.915
TID13	0.859	0.860	0.859	0.921	0.895



(a)LIVE-RMSE (b) MULTI-RMSE (c) TID-RMSE

(d)LIVE-PLCC (e) MULTI-PLCC (f) TID-PLCC

(g) LIVE-SRCC (h) MULTI-SRCC (i) TID-SRCC

Figure 60: Performance of boosting methods versus number of fused methods.

6.2.3.3 Part 2

In this section, we discuss the relative performance change as a consequence of adding more image quality estimators into boosting algorithms. We start with the worst performing quality estimators in each category and add the next best into boosting in the next step. Based on the results in Table 37, we rank the methods for each database in a descending order in the root mean square error category, and in an ascending order in the Pearson and the Spearman correlation categories. The results are given in Fig. 60 in which the lengths of the main bars correspond to the mean values and

the lengths of the thin bars plotted over the main bars show the standard deviations. We plot a horizontal black line in correlation figures, after which the increase in correlation coefficients becomes statistically significant with respect to the regressed worst performing quality estimator. Red bars correspond to the performance of support vector machine-based boosting and blue bars correspond to neural network-based boosting.

As the number of fused image quality estimators increase, there is a general decrease in terms of root mean square error and an increase in terms of the Pearson and the Spearman correlations. Neural network-based boosting outperforms support vector machine-based boosting in terms of root mean square error in all the boosting scenarios when two or more methods are fused. Both Pearson and Spearman follow a non-decreasing behavior with respect to the number of fused methods other than a few exceptions. In terms of the Pearson correlation, neural network-based boosting outperforms support vector machine-based boosting in all the boosting scenarios. In terms of the Spearman correlation, the worst performing quality estimators regressed with support vector machines perform slightly better than quality estimators regressed with neural networks in the LIVE and the MULTI databases. However, in most of the scenarios, neural network-based boosting outperforms support vector machine-based boosting in this experimental setup.

6.2.4 Summary

Existing studies in the literature generally boost methods that are based on fidelity, structure, scale space, and visual system. In addition to these characteristics we also use quality estimators that utilize color information and an alternative boosting strategy [7] as summarized in Table 41. In this thesis and in [7], we analyze the effect of boosting in image quality assessment through multi-method fusion. Existing multi-method studies focus on proposing a single quality estimator. On the contrary, we

Table 41: Characteristics of boosting-based image quality assessment algorithms including the thesis work.

YEAR		2008	2011	2013	2015	2016
EXISTING STUDIES		Liu and Yang	Liu *et al.*			Temel
Fidelity		■	■	■	■	■
Structure		■	■	■	■	■
Scale Space		■	■	■		■
Visual system		■	■	■	■	■
Color					■	■
Boosting Method	SVR		■	■	■	■
	Others	Canonical Correlation Analysis				Neural Networks

investigate the generalizability of multi-method fusion as a framework. In addition to support vector machines that are commonly used in multi-method fusion, we propose using neural networks in boosting. To span different types of image quality assessment algorithms, we use quality estimators based on fidelity, perceptually-extended fidelity, structural similarity, spectral similarity, color, and learning. In the experiments, we perform k-fold cross validation using the LIVE, the MULTI, and the TID13 databases, and the performance of image quality assessment algorithms are measured via accuracy-, linearity-, and ranking-based metrics. Based on the experiments, we show that boosting methods generally improve the performance of image quality assessment. In 17 out of 18 comparisons, boosting-based methods outperform existing best performing methods and the level of enhancement depends on the type of the boosting algorithm. Our experimental results also indicate that boosting the worst performing quality estimator with two or more additional methods leads to statistically significant performance enhancements independent of the boosting technique and neural network-based boosting outperforms support vector machine-based boosting when two or more methods are fused.

CHAPTER VII

CONCLUSION

7.1 Contributions

We analyze a formulation of contrast sensitivity mechanisms in retinal ganglion cells and a formulation of suppression mechanisms in cortical neurons, and utilize these formulations to complement color-based methods in perceived image quality assessment. We investigate the accuracy of existing methods in terms of quantifying perceived color degradations. We show that pixel-wise chroma fidelity and CIEDE color differences are not reliable in the presence of significant color degradations and propose using color name distances for perceived image quality assessment. To the best of our knowledge, color names were not used in the existing image quality assessment methods. We introduce two image quality assessment methods that generally outperform existing methods. Unlike the majority of existing methods, we utilize color information in both of the introduced methods `PerSIM` [1] and `CSV` [2]. We introduce the biologically inspired spatiochromatic similarity method `BLeSS` [3] to assist existing image quality assessment methods that originally oversimplify the role of color in perception. We show that BLeSS-assistance leads to statistically significant enhancements in color-based degradations.

We propose estimating perceived image quality using sparse representations obtained from generic image databases through an unsupervised learning approach. To the best of our knowledge, the introduced method `UNIQUE` [4] is the only quality estimator based on comparing the monotonicity of sparse representations, which does not require subjective scores or distorted images in the training phase. Moreover, we also introduce an extended version of `UNIQUE` as `MS-UNIQUE` [5], which builds on `UNIQUE`

by scaling the weight set based on the sharpness and by using multiple learning architectures to represent image patches through different abstraction layers. `MS-UNIQUE` is further extended as `DMS-UNIQUE` by pooling the projected features that lead to maximum activations and adding an extra layer with multiple linear decoders. We also formulate how linear decoders can be stacked in the weight set generation phase to obtain deeper architectures. The high performance of `UNIQUE` and its extensions show that unsupervised learning-based sparse representations, which do not require distortion specific data or subjective opinions in the training, can robustly estimate perceived quality.

We perform a comparative study of quality and content-based spatial pooling strategies in image quality assessment. The conducted study includes 3 different quality attributes, 3 different image quality databases, more than $4,000$ images, 7 distortion categories, and 7 spatial pooling strategies. Based on the experimental studies, we show that even spatial pooling strategy selection can lead to performance changes, quality attribute selection is still the key in determining the performance, especially when generic pooling strategies are used over pre-designed quality attributes. Also, we observe that it is easier to differentiate the performance of pooling strategies and quality attributes as the number of distortion types increase in the validation set. To the best of our knowledge, there is not any published comparative study of spatial pooling strategies that is comparable to the size of the conducted study in this thesis.

We analyze the effect of boosting in image quality assessment through multi-method fusion. In contrast to existing studies that propose a single quality estimator, we investigate the generalizability of multi-method fusion as a framework. In addition to support vector machines that are commonly used in multi-method fusion studies, we propose using neural-networks in boosting. Based on experimental studies, we show that boosting methods generally improve the performance of image quality assessment and the level of improvement depends on the type of the

boosting algorithm. Our experimental results also indicate that boosting the worst performing quality estimator with two or more methods leads to statistically significant performance enhancements independent of the boosting technique and neural network-based boosting outperforms support vector machine-based boosting when two or more methods are fused.

In this section, we summarize our direct contributions to the literature. In addition to these direct contributions, we also want to emphasize high level findings of this thesis. To understand and measure perceived quality, the best example is our visual system and we should model it as much as we can. Color perception must be included in a comprehensive visual system model. Hand-crafting is not sufficient to obtain comprehensive image quality estimators, we should also learn from the data. Labels are not easy to find in the training. Therefore, we need to focus more on unsupervised approaches. The visual representations that we obtain from handcrafted or data-driven methods need to be pooled to obtain a quality score. Instead of a static pooling strategy that is used for all visual representations, we need to design adaptive pooling strategies that depends on what we perceive. In addition to designing quality maps and pooling strategies, we should also work on boosting existing methods to obtain superior quality estimators. Each existing image quality estimator can be good at capturing specific degradations and we can combine these estimators in a complementary way with boosting. In this thesis, we concentrate of visual representations based on spatial characteristics. However, we also need to utilize temporal, depth-based, and other characteristics to measure perceived quality for other formats and platforms including videos, 3D, virtual and augmented reality.

7.2 Prospective Research Directions

Comprehensive visual system models are the key to accurately measure perceived quality. In addition to designing visual representations based on these comprehensive

models, we also need to design smarter spatial pooling strategies. To enhance the validation of visual representations, we need more informative subjective opinions as ground truths. Crowdsourcing should be used to obtain large scale validation sets with ground truths and these sets can be used to obtain quality estimators based on deep learning. In addition to the quality of images, we also need to consider the aesthetics to comprehensively measure the quality of experience.

To model the perception of colors, perceptual color differences and center-surround-based models are promising enhancements compared to pixel-wise fidelity approaches. However, none of these approaches are sufficient to model the perception of colors. Perception of structures is well-studied compared to color perception but these structural methods are not accurate models of vision as well. To design more comprehensive human visual system-based perception models, a collaboration is necessary among engineers, visual psychophysicists, neuroscientists, cognitive scientists, and related experts.

In Chapter 2, we show that most of the existing image quality estimators contain a weighted sum or an average operation, which transforms attribute maps to objective quality values. In Chapter 6, we describe various spatial pooling strategies, which overlook local information and do not change according to the characteristics of stimuli. A prospective research direction would be designing smarter pooling strategies that are adaptive.

The databases used in verification and validation of the quality estimators are described in Chapter 4. In these databases, subjective scores are provided as ground truth. However, we actually do not know what the end users perceive. This kind of database generation process oversimplifies the perceived quality and leads to loss of local information by mapping the whole experience into a single number. Instead of relying on subjective scores, some of the recent studies [133, 134, 135, 136, 137] focused on measuring the perceived quality of multimedia using EEG signals. Even

EEG-based methods seem promising, they need significant improvement to become a standard quality assessment method. The more data we can get from the subjects related to the quality, the better we can mimic the perception mechanisms. Therefore, one of the prospective research directions in perceptual quality assessment is to enhance the validation databases, in which we should have more than one final score per image as subjective opinion.

Subjective experiments are usually conducted in controlled environments, where screen, lighting, ambiance, position of the subject, and related factors are adjusted carefully. Moreover, subjective experiments are designed so that it would be long enough to get sufficient data and short enough to avoid distraction. Since it is not easy to satisfy all the requirements to perform subjective tests, there are only a few comprehensive validation databases in the literature. Subjective quality assessment experiments are designed to measure the quality of experience for the end user. However, in real life, we can not put constrains on how subjects can exactly use their devices. Moreover, in the majority of existing databases, experimenters are usually college or graduate students who only represent a specific group in a society. Therefore, in order to model the quality of experience for a standard user and an environment, it would be better to ease the requirements in the test setup. Recently, crowdsourcing-based subjective quality assessment methods [138, 139] started to emerge. These methods enable researchers to obtain generalizable results independent of specific setup or subject requirements. Moreover, big data that comes with crowdsourcing approaches can also lead to more successful training of learning-based methods, especially the methods based on deep architectures.

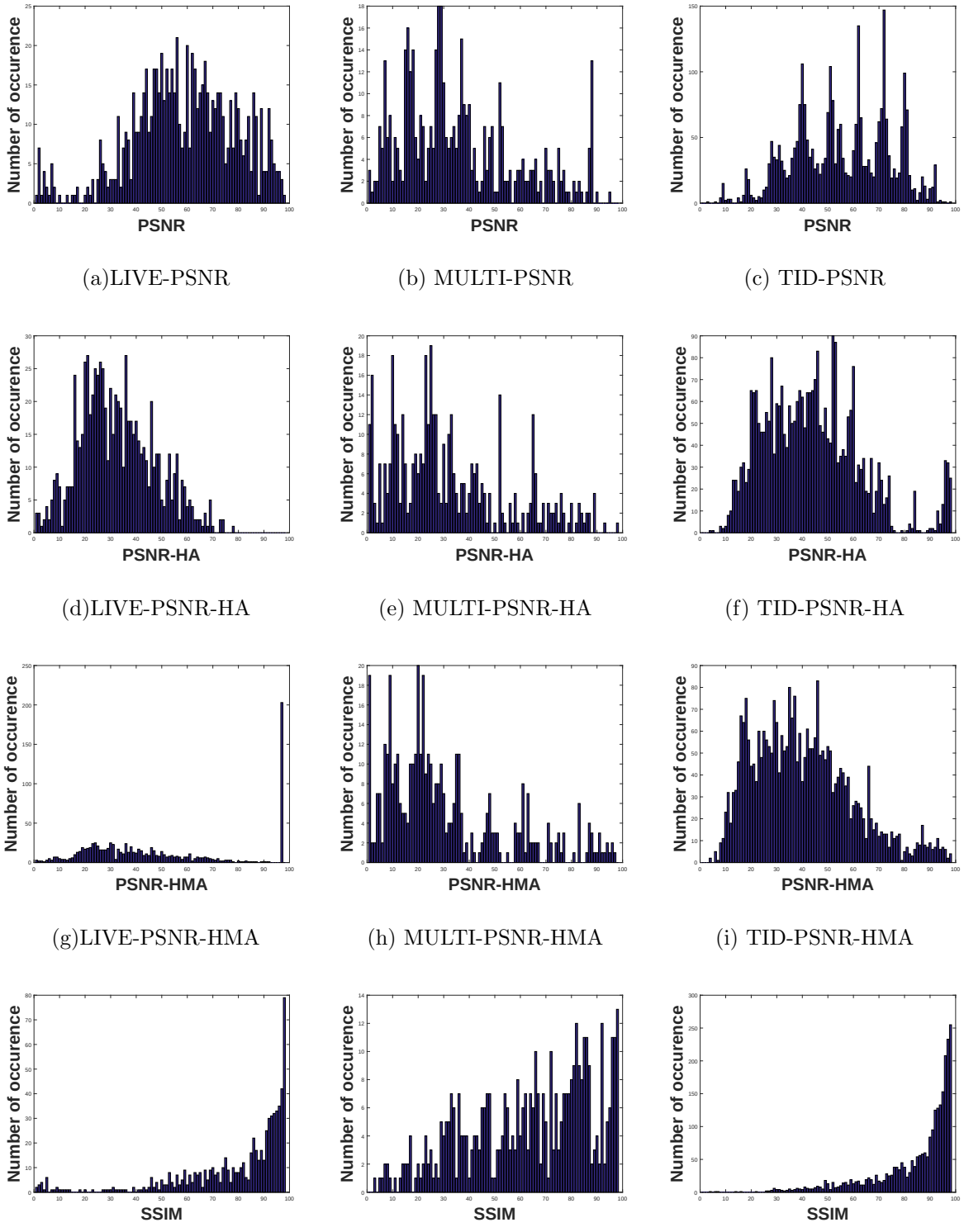
Instead of handcrafting quality attributes, neural network-based approaches [140] are used in the literature to assess the image quality. In addition to basic neural networks, convolutional neural networks [22, 55] are also used to perform learning-based quality assessment. Learning-based methods were not preferred in the past because of

the limited size of existing databases. However, the scale of the databases are changing. RAPID [141] is a deep learning-based image aesthetic assessment model, which was trained and tested using the AVA [142] database that has more than 250,000 images. Crowdsourcing leads to the generation of such databases and the size of these databases enables successful training of deep architectures. Therefore, the role of deep learning-based methods in perceived quality assessment will get more significant.

The majority of objective visual quality metrics try to estimate perceived quality of images by quantifying degradations. However, these metrics are not capable of measuring the aesthetic value. Aesthetic-based metrics were also developed in the literature [143, 144, 145, 146, 147, 148, 149, 150] but these metrics solely focused on classifying images or videos as high or low quality. Recently, various studies in the literature [151, 141] took the aesthetic quality assessment one step further and designed metrics that can estimate the aesthetic quality score. Aesthetic quality assessment was also used in the literature [152] to automatically enhance the aesthetic quality of images. The progress in aesthetic quality assessment is promising but existing studies are far away from representing a comprehensive model. Moreover, quality assessment and aesthetic assessment problems are still investigated separately. Therefore, as a prospective research field, the perception of aesthetics should be understood and modeled more comprehensively, and aesthetics should be combined with quality while modeling perception. The definition of aesthetics or quality depends on the application and technology. Therefore, we need to update our algorithms to catch up with the technological developments including virtual and augmented reality.

APPENDIX A

NORMALIZED HISTOGRAMS OF OBJECTIVE QUALITY SCORES



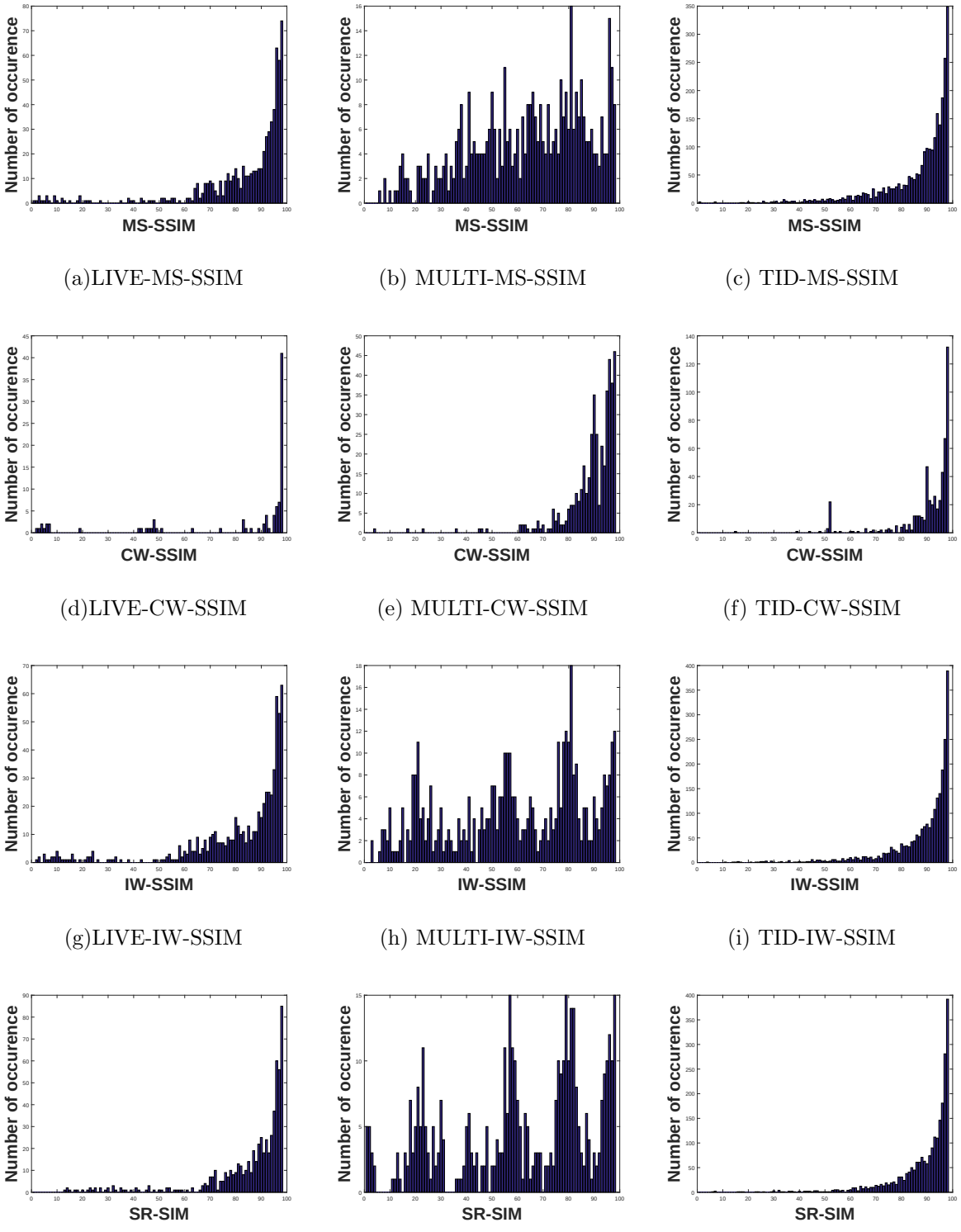
(a)LIVE-PSNR (b) MULTI-PSNR (c) TID-PSNR

(d)LIVE-PSNR-HA (e) MULTI-PSNR-HA (f) TID-PSNR-HA

(g)LIVE-PSNR-HMA (h) MULTI-PSNR-HMA (i) TID-PSNR-HMA

(j)LIVE-SSIM (k) MULTI-SSIM (l) TID-SSIM

Figure 61: Normalized histograms of objective quality estimates PSNR, PSNR-HA, PSNR-HMA, and SSIM.



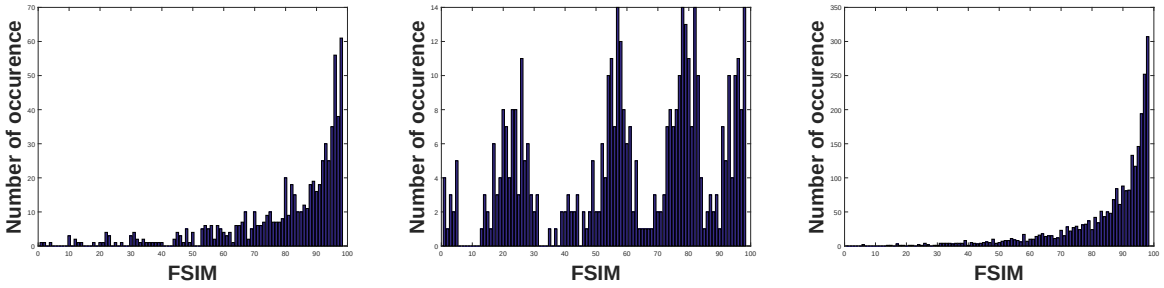
(a)LIVE-MS-SSIM (b) MULTI-MS-SSIM (c) TID-MS-SSIM

(d)LIVE-CW-SSIM (e) MULTI-CW-SSIM (f) TID-CW-SSIM

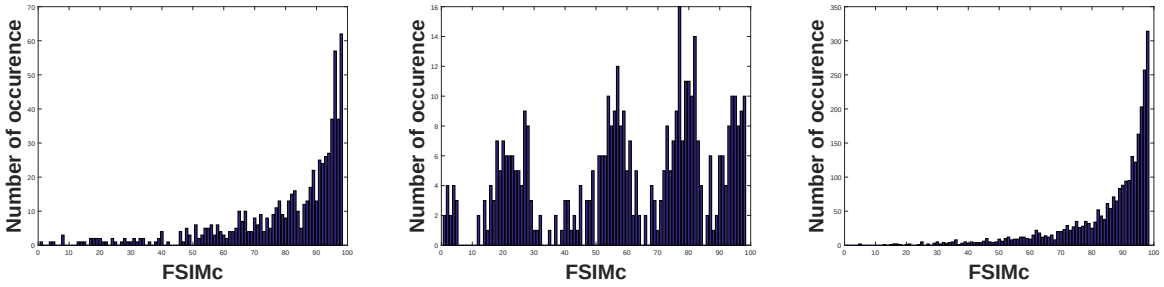
(g)LIVE-IW-SSIM (h) MULTI-IW-SSIM (i) TID-IW-SSIM

(j)LIVE-SR-SIM (k) MULTI-SR-SIM (l) TID-SR-SIM

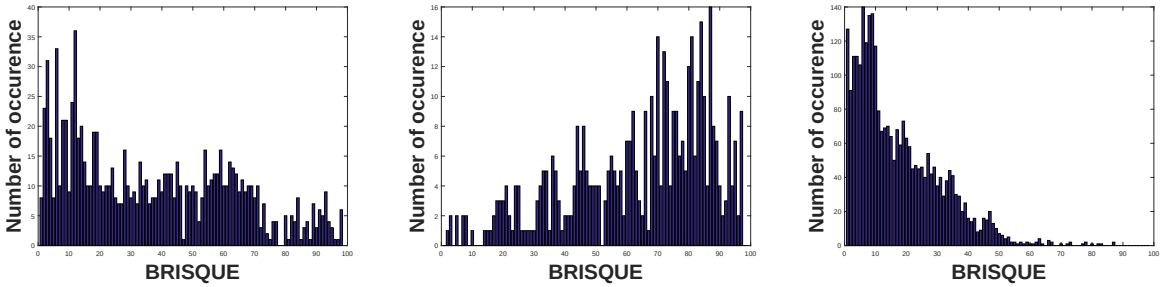
Figure 62: Normalized histograms of objective quality estimates MS-SSIM, CW-SSIM, IW-SSIM, and SR-SIM.



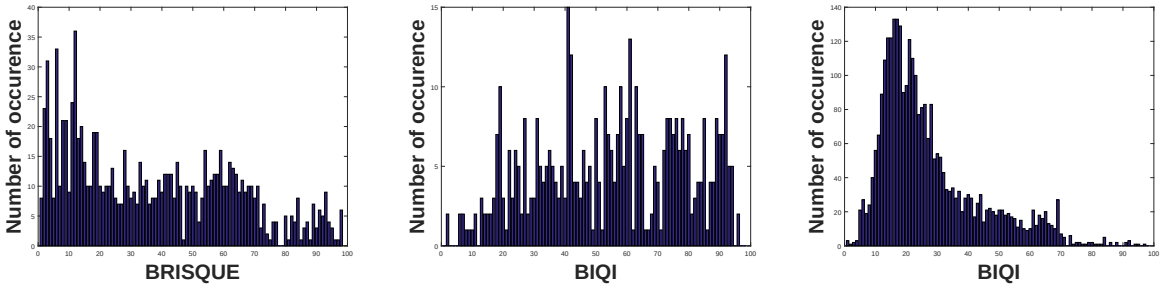
(a)LIVE-FSIM (b) MULTI-FSIM (c) TID-FSIM



(d)LIVE-FSIMc (e) MULTI-FSIMc (f) TID-FSIMc

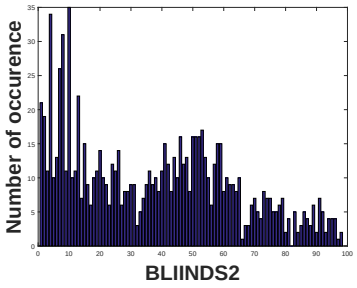


(g)LIVE-BRISQUE (h) MULTI-BRISQUE (i) TID-BRISQUE

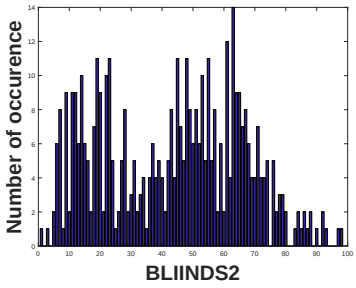


(j)LIVE-BRISQUE (k) MULTI-BIQI (l) TID-BIQI

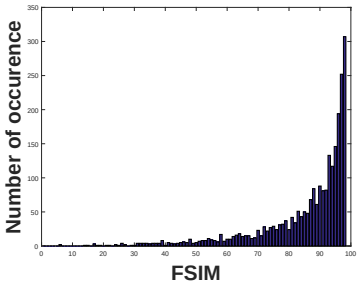
Figure 63: Normalized histograms of objective quality estimates FSIM, FSIMc, BRISQUE, and BIQI.



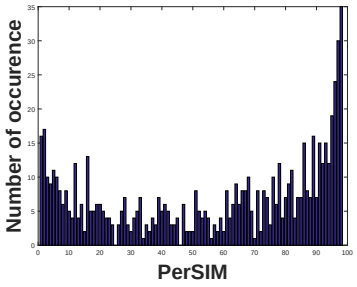
(a)LIVE-BLIINDS2



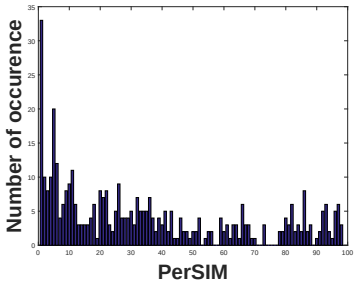
(b) MULTI-BLIINDS2



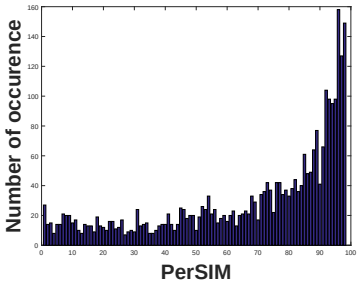
(c) TID-BLIINDS2



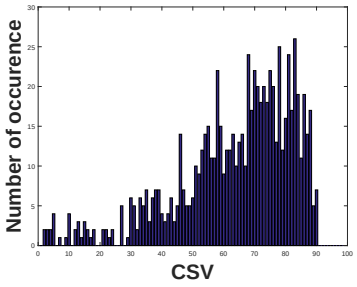
(d)LIVE-PerSIM



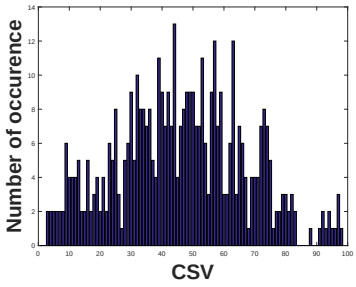
(e) MULTI-PerSIM



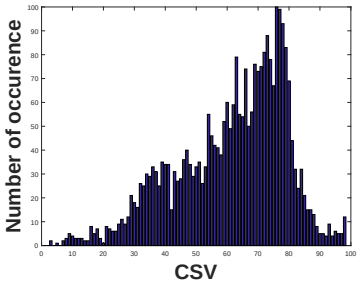
(f) TID-PerSIM



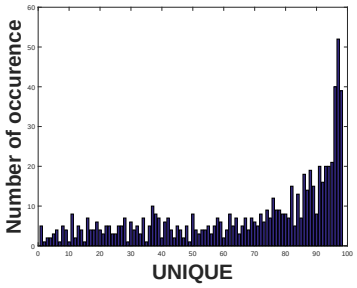
(g)LIVE-CSV



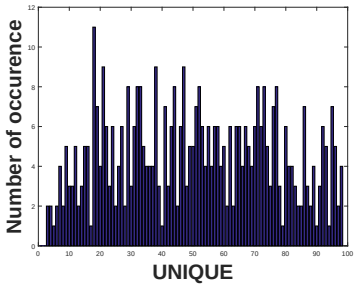
(h) MULTI-CSV



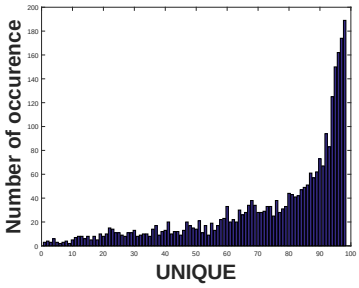
(i) TID-CSV



(j)LIVE-UNIQUE



(k) MULTI-UNIQUE



(l) TID-UNIQUE

Figure 64: Normalized histograms of objective quality estimates Bliinds2, PerSIM, CSV, and UNIQUE.

REFERENCES

[1] D. Temel and G. AlRegib, "PerSIM: Multi-resolution image quality assessment in the perceptually uniform color domain," in *2015 IEEE International Conference on Image Processing (ICIP)*, Sept 2015, pp. 1682–1686.

[2] D. Temel and G. AlRegib, "CSV: Image quality assessment based on color, structure and visual system ," *Signal Processing: Image Communication*, vol. 48, pp. 92 – 103, 2016.

[3] D. Temel and G. Alregib, "BLeSS: Bio-inspired low-level spatiochromatic similarity assisted image quality assessment ," *IEEE International Conference on Multimedia and Expo*, 2016.

[4] D. Temel, M. Prabhushankar, and G. AlRegib, "UNIQUE: Unsupervised image quality estimation," *IEEE Signal Processing Letters*, vol. 23, no. 10, pp. 1414–1418, Oct 2016.

[5] M. Prabhushankar, D. Temel, and G. AlRegib, "MS-UNIQUE: Multi-model and Sharpness-weighted Unsupervised Image Quality Estimation," *Electronic Imaging, Image Quality and System Performance XIV*, 2017.

[6] D. Temel and G. AlRegib, "A comparative study of quality and content-based spatial pooling strategies in image quality assessment," in *2015 IEEE Global Conference on Signal and Information Processing (GlobalSIP)*, Dec 2015, pp. 732–736.

[7] D. Temel and G. Alregib, "Boosting in image quality assessment," *Workshop on Multimedia Signal Processing*, 2016.

[8] K. Morrison, "How many photos are uploaded to Snapchat every second?," 2015, http://www.adweek.com/socialtimes/how-many-photos-are-uploaded-to-snapchat-every-second/621488.

[9] LG Electronics, "Ultra clarity, ultra scale," 2016, http://www.lg.com/hk_en/ultraHD/index.

[10] M. Zhang, "Instagram resolution increase: Here's how it affects image quality and file size," 2015, http://petapixel.com/2015/07/08/instagram-resolution-increase-heres-how-it-affects-image-quality-and-file-size/.

[11] L. Zhang, L. Zhang, X. Mou, and D. Zhang, "FSIM: A Feature similarity index for image quality qssessment.," *IEEE Transactions on Image Processing*, vol. 20, no. 8, pp. 2378–86, Aug. 2011.

[12] D. A. Silverstein and J. E. Farrell, "The relationship between image fidelity and image quality," in *International Conference on Image Processing*, Sep 1996, vol. 1, pp. 881–884 vol.1.

[13] Iain E. Richardson, *Video codec design: Developing image and video compression systems*, John Wiley & Sons, Inc., New York, NY, USA, 2002.

[14] Z. Wang, E. P. Simoncelli, and A. C. Bovik, "Multi-scale structural similarity for image quality assessment," *the Thirty-Seventh Asilomar Conference on Signals, Systems and Computers*, vol. 2, pp. 9–13, 2004.

[15] Z. Wang and E. P. Simoncelli, "Translation insensitive image similiarity in complex wavelet domain," vol. II, no. March, pp. 573–576, 2005.

[16] Z. Wang and Q. Li, "Information content weighting for perceptual image quality assessment.," *IEEE Transactions on Image Processing*, vol. 20, no. 5, pp. 1185–98, May 2011.

[17] K. Egiazarian, J. Astola, N. Ponomarenko, V. Lukin, F. Battisti, and M. Carli, "A new full-reference quality metrics based on HVS," in *the proceedings of VPQM,*, 2006.

[18] N. Ponomarenko, F. Silvestri, K. Egiazarian, M. Carli, J. Astola, and V. Lukin, "On between-coefficient contrast masking of DCT basis functions," in *the proceedings of VPQM*, 2007, pp. 1–4.

[19] N. Ponomarenko, O. Ieremeiev, V. Lukin, K. Egiazarian, and M. Carli, "Modified image Visual quality metrics for contrast change and mean shift accounting," *the proceedings of CADSM*, 2011.

[20] Mathieu Carnec, Patrick L E Callet, and Dominique Barba, "Objective quality assessment of color images based on a generic perceptual reduced reference," vol. 4, no. April, pp. 239–256, 2008.

[21] A. Mittal, A. K. Moorthy, and A. C. Bovik, "No-reference image quality assessment in the spatial domain.," *IEEE Transactions on Image Processing*, vol. 21, no. 12, pp. 4695–708, Dec. 2012.

[22] Le Kang, Peng Ye, Yi Li, and David Doermann, "Convolutional Neural Networks for No-Reference Image Quality Assessment," *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 1733– 1740, 2014.

[23] Q. Li and Z. Wang, "Reduced-reference image quality assessment using divisive normalization-based image representation," *IEEE Journal of Selected Topics in Signal Processing*, vol. 3(2), pp. 202–211, 2009.

[24] S. A. Golestaneh and L. J. Karam, "Reduced-reference quality assessment based on entropy of DNT coefficients of locally weighted Gradients," *IEEE International Conference on Image Processing (ICIP)*, 2015.

[25] C. Blakemore and F. W. Campbell, "On the existence of neurones in the human visual system selectively sensitive to the orientation and size of retinal images," in *The Journal of Physiology*, 1969, vol. 203(1), pp. 237–260.

[26] Y. Yang, J. Ming, and N. Yu, "Color image quality assessment based on CIEDE2000," *Advances in Multimedia*, vol. 2012, pp. 1–6, 2012.

[27] Christian J. V. Lambrecht, *Vision models and applications to image and video processing* , Kluwer Academic Publishers, 2001.

[28] T. Kinsman, M. Fairchild, and J. Pelz, "Color is not a metric space," *Implications for Pattern Recognition, Machine Learning, and Computer Vision*, 1980.

[29] CIE, "Improvement to industrial colour-difference evaluation," in *Vienna: CIE Publication No. 142-2001*, 2001.

[30] M. Luo and G. Cui and B. Rigg, "The development of the CIE 2000 colour-difference formula: CIEDE2000," *Color Research & Application*, 2001.

[31] Sharma, G. and Wu, W. and Dalal,E., "The CIEDE2000 color-difference formula: implementation notes, supplementary test data and mathematical observations," *Color Research & Application*, 2005.

[32] X. Zhang, D.A. Silverstein, J.E. Farrell, and B.A. Wandell, "Color image quality metric S-CIELAB and its application to halftone texture visibility," in *IEEE International Computer Conference*, 1997.

[33] G. M. Johnson, "Measuring images : Differences , quality and appearance," *Proc. of SPIE Electronic Imaging Conference*, 2003.

[34] D. M. Chandler and S. S. Hemami, "VSNR: A Wavelet-based visual signal-to-noise ratio for natural images.," *IEEE Transactions on Image Processing*, vol. 16, no. 9, pp. 2284–98, Sept. 2007.

[35] L. Zhang and H. Li, "SR-SIM: A Fast and high performance IQA index based on spectral residual," in *Image Processing (ICIP), 2012 19th IEEE International Conference on*, Sept 2012, pp. 1473–1476.

[36] N. Damera-Venkata, T. D. Kite, W. S. Geisler, B. L. Evans, and A. C. Bovik, "Image quality assessment based on a degradation model.," *IEEE Transactions on Image Processing*, vol. 9, no. 4, pp. 636–50, Jan. 2000.

[37] M. J. Wainwright, O. Schwartz, and E. P. Simoncelli, "Natural image statistics and divisive normalization: Modeling nonlinearity and adaptation in cortical neurons," in *Probabilistic Models of the Brain: Perception and Neural Function*, R Rao, B Olshausen, and M Lewicki, Eds., chapter 10, pp. 203–222. MIT Press, Feb 2002.

[38] E. P. Simoncelli, W. T. Freeman, E. H. Adelson, and D. J. Heeger, "Shiftable multi-scale transforms," *IEEE Transactions on Information Theory*, vol. 38(2), pp. 587–607, 1992.

[39] E.P. Simoncelli, "Statistical models for images: compression, restoration and synthesis," in *Thirty-First Asilomar Conference on Signals, Systems, and Computers*, Nov 1997, vol. 1, pp. 673–678.

[40] E. P. Simoncelli, "Cortical normalization models and the statistics of visual images," in *Neural Information and Coding Workshop*, 1999.

[41] M. J. Wainwright and E. P. Simoncelli, "Scale mixtures of gaussians and the statistics of natural images," *Neural Information Processing Systems(NIPS)*, vol. 12, pp. 855–861, 1999.

[42] C. Enroth-Cugell and J. G. Robson, "The contrast sensitivity of Retinal Ganglion cells of the cat," *The Journal of Physiology*, 1966.

[43] R. A. Young, "The Gaussian derivative model for spatial vision: I. Retinal mechanisms," *Spatial Vision*, 1987.

[44] H.R. Sheikh and A.C. Bovik, "Image information and visual quality," *IEEE Transactions on Image Processing*, vol. 15, no. 2, pp. 430–444, Feb. 2006.

[45] M.P. Sampat, Zhou Wang, S. Gupta, A.C. Bovik, and M.K. Markey, "Complex wavelet structural similarity: A new image similarity index," *Image Processing, IEEE Transactions on*, vol. 18, no. 11, pp. 2385–2401, Nov 2009.

[46] N. Damera-Venkata, T.D. Kite, W.S. Geisler, B.L. Evans, and A.C. Bovik, "Image quality assessment based on a degradation model," *IEEE Transactions on Image Processing*, vol. 9, no. 4, pp. 636–650, Apr 2000.

[47] Huixuan Tang, Neel Joshi, and Ashish Kapoor, "Learning a blind measure of perceptual image quality," *Cvpr 2011*, vol. 1, pp. 305–312, jun 2011.

[48] H. Tang, N. Joshi, and A. Kapoor, "Blind Image Quality Assessment Using Semi-supervised Rectifier Networks," *2014 IEEE Conference on Computer Vision and Pattern Recognition*, pp. 2877–2884, jun 2014.

[49] A. K. Moorthy and A. C. Bovik, "Blind image quality assessment: From natural scene statistics to perceptual quality," *IEEE Transactions on Image Processing*, vol. 20, no. 12, pp. 3350–3364, Dec 2011.

[50] Peng Ye, Jayant Kumar, David Doermann, and Le Kang, "Unsupervised feature learning framework for no-reference image quality assessment," *Computer Vision and Pattern Recognition (CVPR), 2012 IEEE Conference on*, pp. 1098–1105, 2012.

[51] Peng Ye, Jayant Kumar, Le Kang, and David Doermann, "Real-time no-reference image quality assessment based on filter learning," *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, pp. 987–994, 2013.

[52] L. Zhang, Z. Gu, X. Liu, H. Li, and J. Lu, "Training quality-aware filters for no-reference image quality assessment," *IEEE MultiMedia*, vol. 21, no. 4, pp. 67–75, Oct 2014.

[53] T. Guha, E. Nezhadarya, and R. K. Ward, "Sparse Representation-based Image Quality Assessment," vol. 6, no. 1, pp. 1–10, 2013.

[54] L. Kang, P. Ye, Li Yi, and D. Doermann, "Simultaneous Estimation of Image Quality and Distortion via Multi-task Convolutional Neural Networks," in *The International Conference on Image Processing (ICIP 2015)*, September 2015.

[55] J. Li, L. Zou, J. Yan, D. Deng, T. Qu, and G. Xie, "No-reference image quality assessment using Prewitt magnitude based on convolutional neural networks," *Signal, Image and Video Processing*, vol. 10, no. 4, pp. 609– 616, 2015.

[56] S. Bianco, L. Celona, P. Napoletano, and R. Schettini, "On the Use of Deep Learning for Blind Image Quality Assessment," pp. 1–7.

[57] J. Deng, W. Dong, R. Socher, L.-J. Li, K. Li, and L. Fei-Fei, "ImageNet: A Large-Scale Hierarchical Image Database," in *CVPR09*, 2009.

[58] Bolei Zhou, Agata Lapedriza, Jianxiong Xiao, Antonio Torralba, and Aude Oliva, "Learning deep features for scene recognition using places database," in *Advances in Neural Information Processing Systems 27*, Z. Ghahramani, M. Welling, C. Cortes, N. D. Lawrence, and K. Q. Weinberger, Eds., pp. 487–495. Curran Associates, Inc., 2014.

[59] Christophe Charrier, Olivier Lézoray, and Gilles Lebrun, "Machine learning to design full-reference image quality assessment algorithm," *Signal Processing: Image Communication*, vol. 27, no. 3, pp. 209–219, mar 2012.

[60] Q. Sang, X. Wu, C. Li, and A. C. Bovik, "Blind image quality assessment using a reciprocal singular value curve," *Signal Processing: Image Communication*, vol. 29, no. 10, pp. 1149 – 1157, 2014.

[61] Wufeng Xue, Lei Zhang, and Xuanqin Mou, "Learning without Human Scores for Blind Image Quality Assessment," *2013 IEEE Conference on Computer Vision and Pattern Recognition*, pp. 995–1002, jun 2013.

[62] D. Martin, C. Fowlkes, D. Tal, and J. Malik, "A database of human segmented natural images and its application to evaluating segmentation algorithms and measuring ecological statistics," in *Computer Vision, 2001. ICCV 2001. Proceedings. Eighth IEEE International Conference on*, 2001, vol. 2, pp. 416–423 vol.2.

[63] W. Lu, N. Mei, F. Gao, L. He, and X. Gao, "Blind image quality assessment via semi-supervised learning and fuzzy inference," *Applied Informatics*, vol. 2, no. 1, pp. 9, 2015.

[64] Weilong Hou, Xinbo Gao, Senior Member, Dacheng Tao, and Senior Member, "Blind Image Quality Assessment via Deep Learning," vol. 26, no. 6, pp. 1275–1286, 2015.

[65] Fei Gao, Dacheng Tao, Senior Member, and Xinbo Gao, "Learning to Rank for Blind Image Quality Assessment," *IEEE Trans. on Neural Networks and Learning Systems*, , no. SEPTEMBER 2013, pp. 1–30, 2015.

[66] A. Mittal, A. K. Moorthy, and A. C. Bovik, "Making image quality assessment robust," in *2012 Conference Record of the Forty Sixth Asilomar Conference on Signals, Systems and Computers (ASILOMAR)*, Nov 2012, pp. 1718–1722.

[67] L. He, D. Tao, X. Li, and X. Gao, "Sparse representation for blind image quality assessment," in *Computer Vision and Pattern Recognition (CVPR), 2012 IEEE Conference on*, June 2012, pp. 1146–1153.

[68] Z. Xu, R. Jin, H. Yang, I. King, and M. R. Lyu, "Simple and efficient multiple kernel learning by group lasso," in *Proceedings of the 27th International Conference on Machine Learning (ICML-10)*, Johannes Frnkranz and Thorsten Joachims, Eds. 2010, pp. 1175–1182, Omnipress.

[69] Z. Wang and X. Shang, "Spatial pooling strategies for perceptual image quality assessment," *International Conference on Image Processing*, 2006.

[70] A. K. Moorthy and A. C. the, "Perceptually significant spatial pooling techniques for image quality assessment," *Proc. SPIE, Human Vision and Electronic Imaging XIV*, 2009.

[71] U. Rajashekar, I. van der Linde, A. C. Bovik, and L. K. Cormack, "Gaffe: A gaze-attentive fixation finding engine," *IEEE Transactions on Image Processing*, vol. 17, no. 4, pp. 564–573, 2008.

[72] M. Gong and M. Pedersen, "Spatial pooling for measuring color printing quality attributes," *Journal of Visual Communication and Image Representation*, 2012.

[73] R. Achanta, S. Hemami, F. Estrada, and S. Susstrunk, "Frequency-tuned salient region detection," in *Computer Vision and Pattern Recognition, 2009. CVPR 2009. IEEE Conference on*, 2009, pp. 1597–1604.

[74] H. J. Seo and P. Milanfar, "Nonparametric bottom-up saliency detection by self-resemblance," in *2009 IEEE Computer Society Conference on Computer Vision and Pattern Recognition Workshops*, 2009, pp. 45–52.

[75] H. J. Seo and P. Milanfar, "Static and space-time visual saliency detection by self-resemblance," *Journal of Vision*, vol. 9, no. 12, pp. 15, 2009.

[76] X. Zhang and B. A. Wandell, "A spatial extension of cielab for digital color-image reproduction," *Journal of the Society for Information Display*, vol. 5, no. 1, pp. 61–63, 1997.

[77] G. Simone, C. Oleari, and I. Farup, "Performance of the euclidean color-difference formula in log-compressed osa-ucs space applied to modified-image-difference metrics," in *In 11th Congress of the International Colour Association (AIC*, 2009.

[78] G. Simone, M. Pedersen, J. Y. Hardeberg, and A. Rizzi, "Measuring perceptual contrast in a multi-level framework," vol. 7240.

[79] Z. Wang and J. Y. Hardeberg, "An adaptive bilateral filter for predicting color image difference," in *Color Imaging Conference*, Nov. 2009, pp. 27–31.

[80] Z. Baranczuk, P. Zolliker, and J. Giesen, "Image quality measures for evaluating gamut mapping," in *Color Imaging Conference*, Nov. 2009, pp. 21–26.

[81] C.G. Zewdie and M. Pedersen, "A new pooling strategy for image quality metrics : Five number summary," *EUVIP:5th Europen Workshop on Visual Information Processing*, 2014.

[82] V. Bruni, D. Vitulano, and Z. Wang, "A novel spatial pooling technique for image quality assessment based on luminance-contrast dependence," in *Visual Information Processing (EUVIP), 2014 5th European Workshop on*, Dec 2014, pp. 1–6.

[83] Q. Li, Y.-M. Fang, and J.-T.Xu, "A novel spatial pooling strategy for image quality assessment," *Journal of Computer Science and Technology*, vol. 31, no. 2, pp. 225–234, 2016.

[84] M. Kearns, "Thoughts on hypothesis boosting," *Machine Learning Class Project*, 1988.

[85] M. Kearns and L. Valiant, "Cryptographic limitations on learning boolean formulae and finite automata," *J. ACM*, vol. 41, no. 1, pp. 67–95, Jan. 1994.

[86] R. E. Schapire, "The strength of weak learnability," *Mach. Learn.*, vol. 5, no. 2, pp. 197–227, July 1990.

[87] Mingna Liu and Xin Yang, "A New Image Quality Approach Based on Decision Fusion," *2008 Fifth International Conference on Fuzzy Systems and Knowledge Discovery*, vol. 4, pp. 10– 14, 2008.

[88] T. J. Liu, W. Lin, and C. C. J. Kuo, "A multi-metric fusion approach to visual quality assessment," in *Quality of Multimedia Experience (QoMEX), 2011 Third International Workshop on*, Sept 2011, pp. 72–77.

[89] T. J. Liu, W. Lin, and C. C. J. Kuo, "Image quality assessment using multi-method fusion," *IEEE Transactions on Image Processing*, vol. 22, no. 5, pp. 1793–1807, May 2013.

[90] T. J. Liu, K. H. Liu, J. Y. Lin, W. Lin, and C. C. J. Kuo, "A ParaBoost method to image quality assessment," *IEEE Transactions on Neural Networks and Learning Systems*, vol. PP, no. 99, pp. 1–15, 2015.

[91] A. Leontaris, P. C. Cosman, and A. R. Reibman, "Quality evaluation of motion-compensated edge artifacts in compressed video," *IEEE Transactions on Image Processing*, vol. 16, no. 4, April 2007.

[92] C.-C. J. Kuo H. Ko, R. Song, "A ParaBoost stereoscopic image quality assessment (PBSIQA) system," *arXiv*, 2016.

[93] Nikolay Ponomarenko, Lina Jin, Oleg Ieremeiev, Vladimir Lukin, Karen Egiazarian, Jaakko Astola, Benoit Vozel, Kacem Chehdi, Marco Carli, Federica Battisti, and C.-C. Jay Kuo, "Image database TID2013: Peculiarities, results and perspectives ," *Signal Processing: Image Communication*, vol. 30, pp. 57 – 77, 2015.

[94] H. Kolb, "Simple Anatomy of the Retina," 2006, http://webvision.med.utah.edu/book/part-i-foundations/simple-anatomy-of-the-retina/.

[95] National Eye Institute, "Facts about the cornea and corneal disease," 2016, https://nei.nih.gov/health/cornealdisease.

[96] O. Pele and M. Werman, "Improving perceptual color difference using basic color terms," *arXiv*, pp. 1–14, 2012.

[97] P. Berlin, B.and Kay, "Basic color terms: Their universality and evolution," *University of California Press*, 1969.

[98] J. Van de Weijer, C. Schmid, J. Verbeek, and D. Larlus, "Learning color names for real-world applications.," *IEEE Transactions on Image Processing*, vol. 18, no. 7, pp. 1512–23, July 2009.

[99] Y. Rubner, C. Tomasi, and L. J. Guibas, "The Earth Movers Distance as a metric for image retrieval," *International Journal on Computer Vision*, vol. 40, no. 2, pp. 99–121, 2000.

[100] X. Otazu *et al.*, "Multiresolution Wavelet Framework Models Brigthness Induction Effects," *Vision Resaerch*, vol. 28, pp. 733–751, 2008.

[101] N. Murray *et al.*, "Low-Level Spatiochromatic Grouping for Saliency Estimation," *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, vol. 35, no. 11, pp. 2810–2816, Nov 2013.

[102] B. Blakeslee and M. E. McCourt, "Similar mechanisms underlie simultaneous brightness contrast and grating induction," *Vision Research*, vol. 37, no. 20, pp. 2849 – 2869, 1997.

[103] X. Otazu *et al.*, "Toward a Unified Chromatic Induction Model," *Journal of Vision*, vol. 10(12), pp. 1–24, 2010.

[104] K. T. Mullen, "The contrast sensitivity of human color vision to red-green and blue-yellow chromatic gratings," *The Journal of Physiology*, vol. 359, pp. 381 – 400, 1985.

[105] ITU-T Rec. BT.500-13, "Methodology for the Subjective Assessment of the Quality og Television Pictures," 2012.

[106] H. R. Sheikh *et al.*, "LIVE image quality assessment database release 2," 2006, live.ece.utexas.edu/research/quality/subjective.htm.

[107] D. Jayaraman, A. Mittal, A. K. Moorthy, and A. C. Bovik, "Objective Quality Assessment of Multiply Distorted Images," *Proceedings of Asilomar Conference on Signals, Systems and Computers*, 2012.

[108] ITU-T Rec. BT.500-11, "Methodology for the Subjective Assessment of the Quality og Television Pictures," 2002.

[109] D. H. Brainard, "The Psychophysics Toolbox," 1997.

[110] K. N. Plataniotis and A. N. Venetsanopoulos, "Color Image Processing and Applications," 2010.

[111] K. Dabov, A. Foi, V. Katkovnik, and K. Egiazarian, "Image denoising by sparse 3-d transform-domain collaborative filtering," *IEEE Transactions on Image Processing*, vol. 16, no. 8, pp. 2080–2095, Aug 2007.

[112] N. Ponomarenko, V. Lukin, A. Zelensky, K. Egiazarian, M. Carli, and F. Battisti, "Tid 2008 - a database for evaluation of full-reference visual quality assessment metrics," *Advances of Modern Radioelectronics*, vol. 10, pp. 30–45, 2009.

[113] N. Ponomarenko, V. Lukin, K. Egiazarian, and J. Astola, "ADCTC: Advanced DCT-based Image Coder," 2008.

[114] J. L. Paredes and G. R. Arce, "Compressive sensing signal reconstruction by weighted median regression estimates," *IEEE Transactions on Signal Processing*, vol. 59, no. 6, pp. 2585–2601, June 2011.

[115] A. Danielyan, A. Foi, V. Katkovnik, and K. Egiazarian, "Spatially adaptive filtering as regularization in inverse imaging," 2010.

[116] H. R. Sheikh, M. F. Sabir, and A. C. Bovik, "A statistical evaluation of recent full reference image quality assessment algorithms," *IEEE Transactions on Image Processing*, 2006.

[117] ITU, "Statistical analysis , evaluation and reporting guidelines of quality measurements," *ITU-T Rec P.1401*, 2012.

[118] H. E. Ross and D. J. Murray, ," in *E. H. Weber on the tactile senses*. Taylor & Francis, Erlbaum (UK), 1996.

[119] O. Pele and M. Werman, "Fast and Robust Earth Mover's Distances," in *ICCV*. 2009, pp. 460–467, IEEE Computer Society.

[120] P. Kovesi, "Image features from phase congruency," *Technical Report of Robotics and Vision Resaerch Group*, 1995.

[121] X. Hou and L. Zhang, "Saliency Detection: A Spectral Residual Approach," in *Computer Vision and Pattern Recognition, 2007. CVPR '07. IEEE Conference on*, June 2007, pp. 1–8.

[122] L. Zhang and H. Li, "SR-SIM: A Fast and High Performance IQA Index based on Spectral Residual," 2012.

[123] L Zhang *et al.*, "FSIM: A Feature similarity index for image quality assessment," 2011.

[124] M. Tkalcic and J. F. Tasic, "Colour spaces: perceptual, historical and applicational background," in *EUROCON 2003. Computer as a Tool. The IEEE Region 8*, Sept 2003, vol. 1, pp. 304–308 vol.1.

[125] Y. Dan, J. J. Atick, and R. C. Reid, "Efficient coding of natural scenes in the lateral ganiculate nucleus: Experimental test of a computational theory," *The Journal of Neuroscience*, vol. 16(10), pp. 3351–3362, 1996.

[126] H. Chang and M. Wang, "Sparse correlation coefficient for objective image quality assessment," *Signal Processing: Image Communication*, vol. 26, no. 10, pp. 577 – 588, 2011.

[127] L. Fei-Fei and O. Russakovsky, "Analysis of large-scale visual recognition," in *Bay Area Vision Meeting*, Oct 2013.

[128] B. A. Olshausen and D. J Field, "Sparse coding with an overcomplete basis set: A strategy employed by v1?," *Vision research*, vol. 37, no. 23, pp. 3311–3325, 1997.

[129] D. M. Bradley and J. A. Bagnell, "Differential sparse coding," 2008.

[130] Z. Wang, A. C. Bovik, H. R. Sheikh, and E. P. Simoncelli, "Image quality assessment: From error visibility to structural similarity.," *IEEE Transactions on Image Processing*, vol. 13, no. 4, pp. 600–12, Apr. 2004.

[131] A. K. Moorthy and A. C. Bovik, "A two-step framework for constructing blind image quality indices," *IEEE Signal Processing Letters*, vol. 17, no. 5, pp. 513–516, May 2010.

[132] M. A. Saad, A. C. Bovik, and C. Charrier, "Blind image quality assessment: A natural scene statistics approach in the dct domain," *IEEE Transactions on Image Processing*, vol. 21, no. 8, pp. 3339–3352, Aug 2012.

[133] E. Kroupi, P. Hanhart, J.S. Lee, M. Rerabek, and T. Ebrahimi, "Eeg correlates during video quality perception," *EUSIPCO*, 2014.

[134] S. Scholler et al., "Toward a direct measure of video quality perception using eeg," *IEEE Transanctions on Image Processing*, vol. 21(5), 2012.

[135] L. Acqualagna et al., "Eeg-based classification of video quality perception using steady state visual evoked potentials (ssveps)," *Journal of Neural Engineering*, vol. 12(2), 2015.

[136] L. Lindemann and M. Magnor, "Assessing the quality of compressed images using eeg," *International Conference on Image Processing*, 2011.

[137] A. N. Moldovan, I. Ghergulescu, S. Weibelzahl, and C. H. Muntean, "User-centered eeg-based multimedia quality assessment," *IEEE International Symposium Broadband Multimedia Systems and Broadcasting* , 2013.

[138] F. Ribeiro, D.Florencio, and V. Nascimento, "Crowdsourcing subjective image quality evaluation," 2011, International Conference on Image Processing.

[139] D. Ghadiyaram and A. C. Bovik, "Crowdsourced study of subjective image quality," 2014, Proceedings of Asilomar Conference on Signals, Systems and Computers.

[140] Alexander Havstad, "Image quality assessment using artificial neural networks," , no. June, 2004.

[141] X. Lu, Z. Lin, H. Jin, J. Yang, and J. Z. Wang, "RAPID: Rating pictorial aesthetics using deep learning," *Proceedings of the ACM International Conference on Multimedia*, 2014.

[142] N. Murray and D. Barcelona and L. Marchesotti and F. Perronnin, "AVA : A Large-scale database for aesthetic visual analysis," *Computer Vision and Pattern Recognition*, 2012.

[143] R. Datta, D. Joshi, J. Li, and J. Z. Wang, "Studying aeshetics in photographic images using a computational approach," *Proceedings of the European Conference on Computer Vision*, 2006.

[144] D. Hasler and S. Susstrunk, "Measuring colorfulness in natural images," *Proceedings of the SPIE*, 2003.

[145] G. Yildirim, A. Shaji, and S. Susstrunk, "Estimating beauty ratings of videos using supervoxels," *Proceedings of the 21st ACM International Conference on Multimedia*, 2013.

[146] S. Dhar, V. Ordonez, and T. L. Berg, "High level describable attributes for predicting aesthetics and interestingness," *IEEE Conference on Computer Vision and Pattern Recognition*, 2011.

[147] Y. Luo and X. Tang, "Photo and video quality evaluation: Focusing on the subject," *European Conference on Computer Vision*, 2008.

[148] X. Tang, W. Luo, and X. Wang, "Content-based photo quality assessment," *IEEE Transactions on Multimedia*, vol. 15(8), 2013.

[149] D. Temel and G. AlRegib, "A comparative study of computational aesthetics," *IEEE International Conference on Image Processing*, 2014.

[150] L. Marchesotti, F. Perronnin, D. Larlus, and G. Csurka, "Assessing the aesthetic quality of photographers using generic image descriptors," *IEEE International Conference on Computer Vision*, 2011.

[151] A. K. Moorthy, P. Obrador, and N. Oliver, "Towards computational models of visual aesthetic appeal of consumer videos," *European Conference on Computer Vision*, 2010.

[152] S. Bhattacharya, R. Sukthankar, and M. Shah, "A framework for photo-quality assessment and enhancement based on visual aesthetics," *Proceedings of the International Conference on Multimedia*, 2010.

VITA

Dogancan Temel received an M.S. degree with a minor in Management in 2013, and a PhD degree with a minor in Computer Science in 2016 from the school of Electrical and Computer Engineering in Georgia Institute of Technology, Atlanta. While his studies at Georgia Tech, Mr. Temel worked in the Multimedia and Sensors Lab at the Center for Signal and Information Processing as a Graduate Research Assistant and in Texas Instruments as a Systems Engineering intern. Mr. Temel worked on various projects including perceived image quality assessment, deep learning-based image processing and computer vision, high color range imaging, vital sign monitoring, computational aesthetics, seismic interpretation, 3D reconstruction, streaming, and quality assessment. In general, Mr. Temel's research interests include any topic that focuses on understanding visual representations.